Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ciencias de la Computación e Informática

CI-0131 Diseño de Experimentos

Dr. Igñacio Díaz Oreiro

Grupo 01

Laboratorio de Regresión Lineal

Nathalie Alfaro Quesada, B90221

David Meléndez Aguilar, C04726

I ciclo 2025

PRIMERA PARTE

Se deben cargar los datos de la temporada 2011 de la liga profesional de béisbol que se encuentran en el archivo beisball.csv. Puede usar el siguiente código:

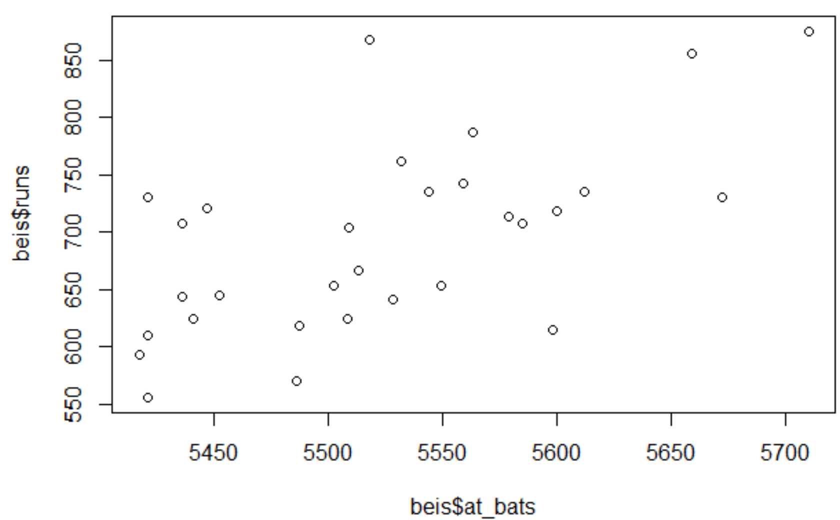
beis = (read.csv(file.choose(), header=T, encoding = "UTF-8")) attach(beis)

Los datos se referencian como “beis”. Además de las carreras anotadas (runs), hay siete variables utilizadas tradicionalmente en el conjunto de datos: at-bats, hits, home runs, batting average, strikeouts, stolen bases, y wins (turnos al bate, hits, jonrones, promedio de bateo, ponches, bases robadas y victorias por lanzador).

También hay tres variables más nuevas: new_onbase, new_slug, new_obs (porcentaje de bateadores que llegan a base, porcentaje de slugging o potencia de bateadores y suma de new_onbase + new_slug), pero estas no las utilizaremos en este trabajo

Se puede usar la función plot() para mostrar la relación entre la variable runs y una de las otras variables numéricas. Trace esta relación usando la variable at_bats como predictor.

Pregunta 1: ¿La relación parece lineal?
Código en R: (Colocar cada línea nueva de código después de la anterior)
plot(beis$at_bats, beis$runs)
Resultado de ejecutar el código anterior:



Análisis:

Aunque no se puede saber a ciencia cierta, podemos observar que los valores se centran entre la esquina inferior izquierda y la esquina superior derecha, por lo que podría parecer lineal.

También se puede cuantificar la fuerza de la relación con el coeficiente de correlación.

Pregunta 2: ¿Qué tan fuerte es la correlación entre runs y at_bats?

Código en R:

```
cor(beis$runs, beis$at_bats)
```

Resultado:

```
> cor(beis$runs, beis$at_bats)
[1] 0.610627
```

Análisis:

Al ser un valor mayor que cero sabemos que es positiva, y al ser mayor que 0.5, podríamos decir que es una correlación fuerte.

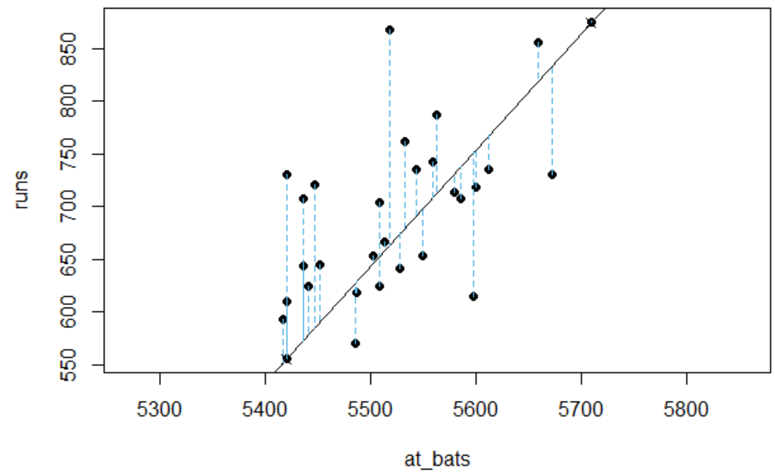
-Suma de residuos al cuadrado

Es la relación de dos variables numéricas, además es útil poder describir la relación de dos variables numéricas, como runs y at_bats, antes mencionadas. Podemos resumir la relación entre estas dos variables encontrando la línea que mejor sigue su asociación. Utilice la siguiente función interactiva para seleccionar la línea (debe hacer click en dos puntos cuando aparezca el gráfico después de ejecutar el código) que cree que hace el mejor trabajo para atravesar la nube de puntos. La línea que especificó se mostrará en negro y los residuos en azul.

Código en R:

```
if(!require('statsr')) {
install.packages('statsr')
library('statsr') }
plot_ss(x = at_bats, y = runs, data = beis)
```

Resultado:




```
> plot_ss(x = at_bats, y = runs, data = beis)

Call:
lm(formula = y ~ x, data = pts)

Coefficients:
(Intercept)            x
  -5429.316        1.104

Sum of Squares:  182576.2
```

Análisis:

En la línea trazada podemos observar que los valores esperados por la línea trazada y los valores observados es bastante variable, siendo algunos de estos relativamente bajos, mientras que otros son más altos.

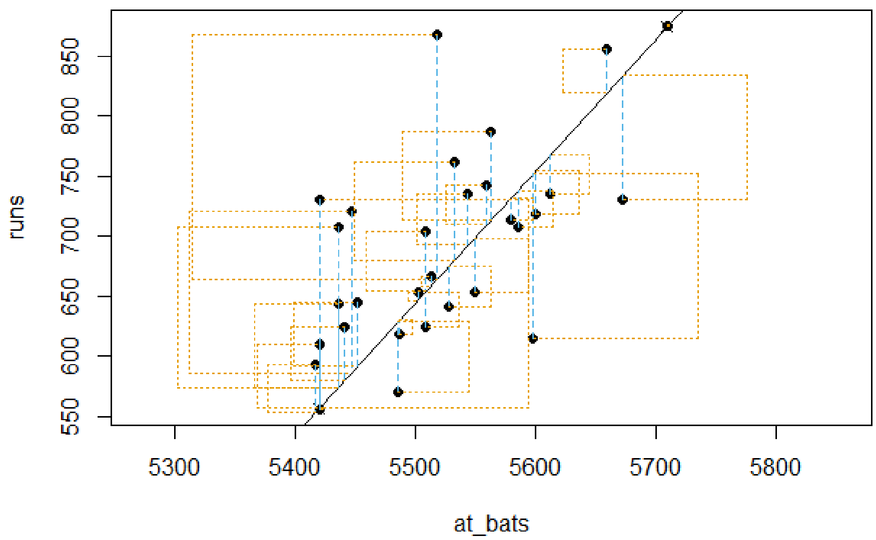
-Residuos cuadrados

Los residuos son la diferencia entre los valores observados y los valores predichos por la línea: ei = yi − ŷi. La forma más común de hacer una regresión lineal es seleccionar la línea que minimiza la suma de los residuos al cuadrado. Para visualizar los residuos cuadrados, puede volver a ejecutar el comando de trazado y agregar el argumento showSquares = TRUE.

Código en R:

plot_ss(x = at_bats, y = runs, data = beis, showSquares = TRUE)

Resultado:




```
> plot_ss(x = at_bats, y = runs, data = beis, showSquares = TRUE)

Call:
lm(formula = y ~ x, data = pts)

Coefficients:
(Intercept)            x
    -5405.5          1.1

Sum of Squares:  180752.7
```

Análisis:

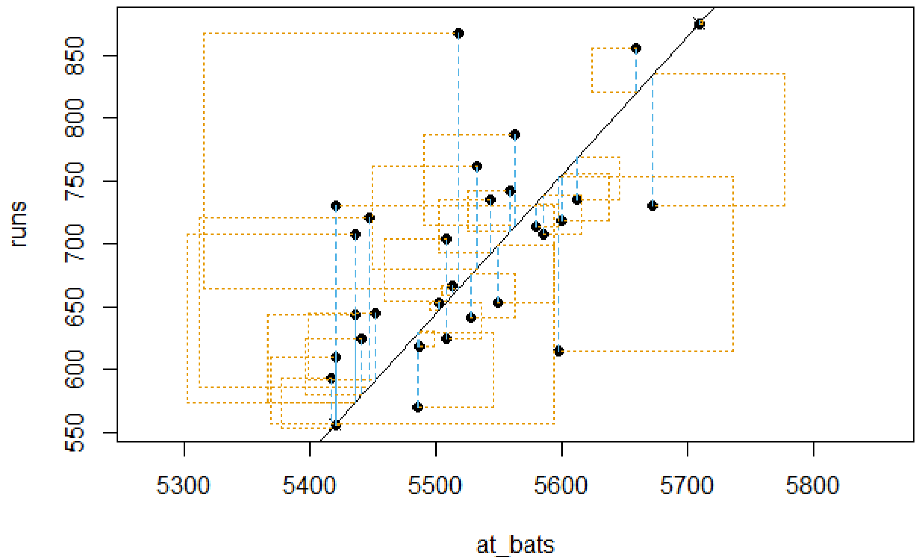
Este nuevo formato nos enseña los residuos en forma de cuadrado, ya que puede resultar más significativo ver el tamaño de los residuos en un cuadrado de dimensión residuo x residuo.

Pregunta 3: Utilizando el código anterior, se ejecuta varias veces con diferentes líneas a trazar, para intentar encontrar una línea que minimice lo más posible la suma de los cuadrados, el menor resultado encontrado fue 180883.5, aunque esto no es preciso, su gráfico es:

Su código en R:

plot_ss(x = at_bats, y = runs, data = beis, showSquares = TRUE)

Resultado:




```
> plot_ss(x = at_bats, y = runs, data = beis, showSquares = TRUE)

Call:
lm(formula = y ~ x, data = pts)

Coefficients:
(Intercept)            x
  -5428.113        1.104

Sum of Squares:  180883.5
```

Análisis:

Se observa que por cada aumento de 1 en at_bats, se predicen 1.104 carreras adicionales. El intercepto (-5428.113) no tiene interpretación directa significativa en este contexto, ya que ningún equipo tiene 0 turnos al bate.

La suma de los cuadrados de los residuos (180883.5) indica el nivel de error del modelo. La visualización de los residuos ayuda a evaluar cómo se ajusta la línea a los datos y si hay patrones de error (como sesgo o varianza desigual).

Modelo lineal

Es difícil obtener la línea correcta de mínimos cuadrados, es decir, la línea que minimiza la suma de los cuadrados de los residuos, a través de prueba y error. En su lugar, se puede usar la función lm en R para ajustar el modelo lineal (también conocido como línea de regresión). La sintaxis en R sería:

m1 <- lm(runs ~ at_bats, data = beis)

El primer argumento en la función lm es la fórmula y ~ x. Aquí se puede leer que se desea hacer un modelo lineal de runs en función de at_bats (variable predictora). La salida de lm es un objeto que contiene toda la información del modelo lineal que se acaba de ajustar. Podemos acceder a esta información mediante la función de summary() en código sería summary(m1).

La salida del summary() consta de varias partes. Primero, la fórmula utilizada para describir el modelo se muestra en la parte superior. Después de la fórmula, aparece información de los residuales. La tabla de “Coeficientes” que se muestra a continuación es clave; su primera columna muestra la intersección y la pendiente del modelo lineal (el coeficiente de at_bats). Con esta tabla, se puede escribir la línea de regresión de mínimos cuadrados para el modelo lineal.

Pregunta 4: Escriba la fórmula del modelo lineal obtenido con los valores correctos de pendiente e intersección con el eje y.

Código en R:

```
m1 <- lm(runs ~ at_bats, data = beis)
summary(m1)
```

Resultado:

```
> summary(m1)

Call:
lm(formula = runs ~ at_bats, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-125.58  -47.05  -16.59   54.40  176.87 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -2789.2429   853.6957  -3.267 0.002871 ** 
at_bats         0.6305     0.1545   4.080 0.000339 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 66.47 on 28 degrees of freedom
Multiple R-squared:  0.3729,	Adjusted R-squared:  0.3505 
F-statistic: 16.65 on 1 and 28 DF,  p-value: 0.0003388
```

Análisis:

Ya que la fórmula del modelo debe tener el formato: $\hat{y}$ = intersección + pendiente $*$ at_bats. Al analizar los resultados, obtenemos los valores estimados de la intersección y la pendiente, por lo que al ingresarlos en la fórmula dada anteriormente, el resultado final sería la siguiente: $\hat{y} = -2789.2429 + 0.6305 *$ at_bats.

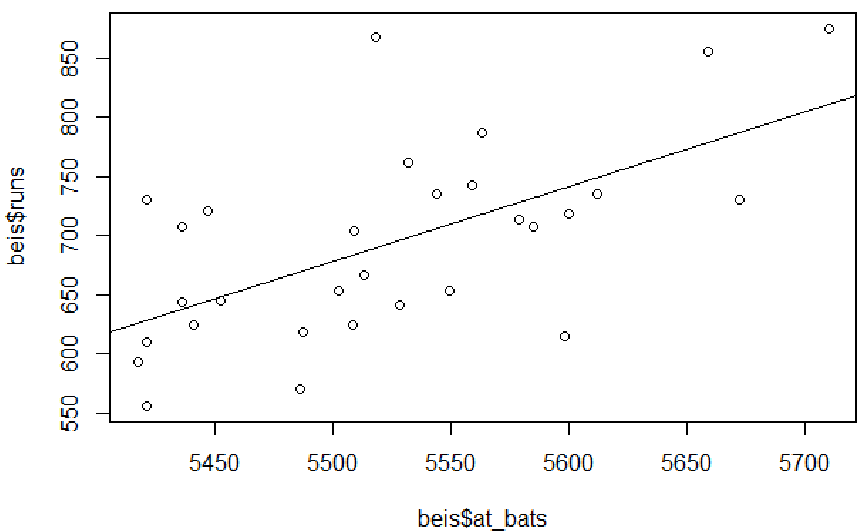
Otro elemento importante del summary() es el Múltiple R-cuadrado, o simplemente, R2 . El valor R2 representa la proporción de variabilidad en la variable de respuesta que es explicada por la variable predictora. Para este modelo, el 37.29% de la variabilidad en runs se explica por at_bats.

En regresión lineal simple, el R2 es el cuadrado del coeficiente de correlación R. Finalmente, el p-value mostrado corresponde a todo el modelo de regresión lineal. Si el p-value es menor que el nivel de significancia (supongamos 0.05) quiere decir que el modelo de regresión lineal predice mejor los resultados que un modelo con solo intercept. Ahora se creará un diagrama de dispersión agregando la línea de mínimos cuadrados.

Código en R:

```
plot(beis$runs ~ beis$at_bats)
abline(m1)
```

Resultado:



Análisis:

La función abline traza una línea basada en su pendiente e intersección. Aquí, se utiliza un atajo al proporcionar el modelo m1, que contiene ambas estimaciones de parámetros. Esta línea se puede usar para predecir “y” en cualquier valor de “x”. Cuando se realizan predicciones para valores de x que están más allá del rango de los datos observados, se denomina extrapolación y, por lo general, no se recomienda. Sin embargo, las predicciones hechas dentro del rango de los datos son más confiables.

Hay una relación positiva entre at_bats y runs. Eso significa que a medida que aumentan los turnos al bate, también tienden a aumentar las carreras anotadas. La pendiente de la línea es positiva, lo cual indica una relación directa. No todos los puntos están sobre la línea, porque hay variabilidad (no todos los equipos tienen la misma eficiencia bateando).

-Comprobación de supuestos

Para evaluar si el modelo lineal es confiable, se debe verificar: la relación lineal de ambas variables, la independencia de residuales, la normalidad de los residuales, y la homogeneidad de varianzas. Sin embargo, en este ejercicio no realizaremos esta validación, aunque en la Pregunta 1 se verificó si la relación entre runs y at_bats era lineal usando un diagrama de dispersión y también se calculó el coeficiente de correlación.

SEGUNDA PARTE

Pregunta 5: Ajuste un nuevo modelo de regresión lineal que use homeruns para predecir runs en vez de at_bats (el anterior). Muestre el código R usado para crear el modelo y la salida summary() de ese modelo.

Código en R:

```
m2 <- lm(runs ~ homeruns, data = beis)
summary(m2)
```

Resultado:

```
> summary(m2)

Call:
lm(formula = runs ~ homeruns, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-91.615 -33.410   3.231  24.292 104.631 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 415.2389    41.6779   9.963 1.04e-10 ***
homeruns      1.8345     0.2677   6.854 1.90e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 51.29 on 28 degrees of freedom
Multiple R-squared:  0.6266,	Adjusted R-squared:  0.6132 
F-statistic: 46.98 on 1 and 28 DF,  p-value: 1.9e-07
```

Análisis:

En este código podemos ver los resultados después de realizar un modelo de regresión lineal que utiliza los homeruns para predecir los runs, donde podemos ver que su p-value nos dice que este predice mejor los resultados que un modelo con solo intercept.

Pregunta 6: Usando los coeficientes del summary() anterior, escriba la ecuación del modelo (la línea de regresión).

Utilizando los resultados de la pregunta anterior, y siguiendo bajo la premisa de que la fórmula: $\hat{y}$ = intersección + pendiente $*$ homeruns. La ecuación modelo, después de reemplazar los valores obtenidos, sería: $\hat{y} = 415.2389 + 1.8345 * \text{homeruns}$.

Pregunta 7: ¿Qué nos dice El Múltiple R-Cuadrado de este modelo con homeruns en comparación con el modelo anterior (para at_bats)? ¿Cuál predice mejor los resultados?

El valor Multiple R-Cuadrado es de 0.6266, lo cuál nos dice que un 62,66% de la variabilidad de runs es explicada por los homeruns, por lo que comparado con el 37,29% obtenida por los at_bats, tenemos que los homeruns predicen mejor los resultados.

Pregunta 8: Ahora que sabe analizar la relación lineal entre dos variables (creando modelos de regresión lineal), investigue las relaciones entre runs y cada una de las variables tradicionales (at_bats, hits, homeruns, bat_avg, strikeouts, stolen_bases, y wins). Ya ha generado la información para at_bats y homeruns.

Calcularemos el coeficiente de correlación de todas las variables predictoras y las mostraremos junto a los modelos. Anteriormente obtuvimos los datos de las relaciones de runs con :

at_bats:

```
> summary(m1)

Call:
lm(formula = runs ~ at_bats, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max
-125.58  -47.05  -16.59   54.40  176.87

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -2789.2429   853.6957  -3.267 0.002871 **
at_bats         0.6305     0.1545   4.080 0.000339 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 66.47 on 28 degrees of freedom
Multiple R-squared:  0.3729,	Adjusted R-squared:  0.3505
F-statistic: 16.65 on 1 and 28 DF,  p-value: 0.0003388
```

homeruns:

```
> summary(m2)

Call:
lm(formula = runs ~ homeruns, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max
-91.615 -33.410   3.231  24.292 104.631

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 415.2389    41.6779   9.963 1.04e-10 ***
homeruns      1.8345     0.2677   6.854 1.90e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 51.29 on 28 degrees of freedom
Multiple R-squared:  0.6266,	Adjusted R-squared:  0.6132
F-statistic: 46.98 on 1 and 28 DF,  p-value: 1.9e-07
```

Y el código utilizado para obtener los coeficientes de correlación, junto con el resultado, es el siguiente:

at_bats:

cor(beis$runs, beis$at_bats)

```
> cor(beis$runs, beis$at_bats)
[1] 0.610627
```

homeruns:

cor(beis$runs, beis$homeruns)

```
> cor(beis$runs, beis$homeruns)
[1] 0.7915577
```

Y seguidamente tenemos los modelos de relación lineal más el código y resultados de los coeficientes de correlación de las variables faltantes:

hits:

```
> summary(m3)

Call:
lm(formula = runs ~ hits, data = beis)

Residuals:
     Min       1Q   Median       3Q      Max 
-103.718  -27.179   -5.233   19.322  140.693 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -375.5600   151.1806  -2.484   0.0192 *  
hits           0.7589     0.1071   7.085 1.04e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 50.23 on 28 degrees of freedom
Multiple R-squared:  0.6419,	Adjusted R-squared:  0.6292 
F-statistic:  50.2 on 1 and 28 DF,  p-value: 1.043e-07
```

cor(beis$runs, beis$hits)

```
> cor(beis$runs, beis$hits)
[1] 0.8012108
```

bat_avg:

```
> summary(m4)

Call:
lm(formula = runs ~ bat_avg, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-94.676 -26.303  -5.496  28.482 131.113 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -642.8      183.1  -3.511  0.00153 ** 
bat_avg       5242.2      717.3   7.308 5.88e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 49.23 on 28 degrees of freedom
Multiple R-squared:  0.6561,	Adjusted R-squared:  0.6438 
F-statistic: 53.41 on 1 and 28 DF,  p-value: 5.877e-08
```

cor(beis$runs, beis$bat_avg)

```
> cor(beis$runs, beis$bat_avg)
[1] 0.8099859
```

strikeouts:

```
> summary(m5)

Call:
lm(formula = runs ~ strikeouts, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-132.27  -46.95  -11.92   55.14  169.76 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1054.7342   151.7890   6.949 1.49e-07 ***
strikeouts    -0.3141     0.1315  -2.389   0.0239 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 76.5 on 28 degrees of freedom
Multiple R-squared:  0.1694,	Adjusted R-squared:  0.1397 
F-statistic: 5.709 on 1 and 28 DF,  p-value: 0.02386
```

cor(beis$runs, beis$strikeouts)

```
> cor(beis$runs, beis$strikeouts)
[1] -0.4115312
```

stolen_bases:

```
> summary(m6)

Call:
lm(formula = runs ~ stolen_bases, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-139.94  -62.87   10.01   38.54  182.49 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  677.3074    58.9751  11.485 4.17e-12 ***
stolen_bases   0.1491     0.5211   0.286    0.777    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 83.82 on 28 degrees of freedom
Multiple R-squared:  0.002914,	Adjusted R-squared:  -0.0327 
F-statistic: 0.08183 on 1 and 28 DF,  p-value: 0.7769
```

cor(beis$runs, beis$stolen_bases)

```
> cor(beis$runs, beis$stolen_bases)
[1] 0.05398141
```

wins:

```
> summary(m7)

Call:
lm(formula = runs ~ wins, data = beis)

Residuals:
     Min       1Q   Median       3Q      Max 
-145.450  -47.506   -7.482   47.346  142.186 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  342.121     89.223   3.834 0.000654 ***
wins           4.341      1.092   3.977 0.000447 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 67.1 on 28 degrees of freedom
Multiple R-squared:  0.361,	Adjusted R-squared:  0.3381 
F-statistic: 15.82 on 1 and 28 DF,  p-value: 0.0004469
```

cor(beis$runs, beis$wins)

```
> cor(beis$runs, beis$wins)
[1] 0.6008088
```

Al realizar la tabla con los valores de correlación, p-value y R^2 obtenemos esto:

Variable predictora	**Correlación**	**p-value del Modelo**	R^2
at_bats	0.610627	0.0003388	0.3729
hits	0.8012108	1.043e-07	0.6419
homeruns	0.7915577	1.9e-07	0.6266
bat_avg	0.8099859	5.877e-08	0.6561
strikeouts	-0.4115312	0.02386	0.1694
stolen_bases	0.05398141	0.7769	0.002914
wins	0.6008088	0.0004469	0.361

Pregunta 9: ¿Cuál de las siete variables anteriores predice mejor la variable runs y por qué lo considera usted así?

La variable que mejor predice la variable runs es la de bat_avg, ya que esta afecta a un 65,61% de todos los casos de runs.

TERCERA PARTE

Ahora realizaremos un análisis de regresión lineal múltiple. Para ello crearemos un modelo de regresión lineal con cinco de las variables originales: at_bats, hits, homeruns, bat_avg, y wins.

Pregunta 10: Con base en el resultado anterior, construya la fórmula del modelo de regresión lineal que corresponde a estas cinco variables. Recuerde que el formato es:

$$\hat{y} = \beta0 + \beta1X1 + \beta2X2 + \ldots + \beta pXp$$

Código en R:

```
mul <- lm(runs ~ at_bats + hits + homeruns + bat_avg + wins, data = beis)
summary(mul)
```

Resultado:

```
> summary(mul)

Call:
lm(formula = runs ~ at_bats + hits + homeruns + bat_avg + wins,
    data = beis)

Residuals:
    Min      1Q  Median      3Q     Max
-48.744 -22.859  -5.446  21.876  59.388

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2583.4214  4006.8878   0.645  0.52521
at_bats        -0.5402     0.7256  -0.745  0.46380
hits            2.2542     2.8344   0.795  0.43424
homeruns        1.0412     0.2442   4.264  0.00027 ***
bat_avg     -9059.2528 15769.7868  -0.574  0.57100
wins            0.8610     0.7611   1.131  0.26914
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 31.3 on 24 degrees of freedom
Multiple R-squared:  0.8808,	Adjusted R-squared:  0.856
F-statistic: 35.47 on 5 and 24 DF,  p-value: 2.504e-10
```

Análisis:

Al obtener estos resultados, y siguiendo la fórmula mencionada en la pregunta, si insertamos los valores obtenidos, la fórmula final es la siguiente:

$\hat{y}$ = 2583.4214 - 0.5402 * at_bats + 2.2542 * hits + 1.0412 * homeruns - 9059.2528 * bat_avg + 0.8610 * wins

Pregunta 11: Según los resultados, ¿el modelo de regresión lineal es significativo? ¿Cómo puede determinarlo?

El modelo de regresión lineal es significativo, debido a que el valor de p-value obtenido es de 2.504e-10, lo que es bastante menor que el umbral puesto de 0.05.

Pregunta 12: Según los resultados del summary(mul), ¿cuánto porcentaje de la variabilidad de la variable de respuesta es atribuible al modelo de regresión lineal? ¿Es este modelo de regresión múltiple un mejor predictor que el mejor modelo de regresión lineal simple de la Pregunta 9?

El porcentaje de variabilidad en la variable de respuesta es obtenido utilizando el valor de Adjusted R-squared, que como puede ser observado en la pregunta 10, es de 0.856, por lo que en porcentaje, este se traduce a un 85.6%. Este resultado, nos da un resultado mejor al obtenido al de la Pregunta 9, ya que al comparar el resultado de la Pregunta 9 de 65,61% con el actual de 85.6%, observamos que el actual es mayor y por lo tanto, mejor.

Pregunta 13:Calcule el Factor de Inflación de la Varianza (VIF) para este modelo múltiple. Presente el código R y los resultados, e indique qué se puede concluir de este cálculo de VIF respecto de la multicolinealidad.

Código en R:

```
library(car)
vif(mul)
```

Resultado:

```
> vif(mul)
  at_bats        hits    homeruns     bat_avg        wins
99.419844 1803.225223    2.234504 1195.442185    2.234290
```

Análisis:

De esto, podemos concluir que hits, bat_avg y at_bats, al tener un VIF bastante alto, tienen una multicolinealidad bastante alta, por lo que la variables están bastante correlacionadas entre ellas, mientras que homeruns y wins tienen un VIF bastante menor, pero que sigue siendo aceptable y con una multicolinealidad moderada.

Pregunta 14: Construya una tabla donde indique las combinaciones de las cuatro variables utilizadas, así como el valor-p del modelo completo, el R2 ajustado y los VIF de esas 4 variables.

Para esta tabla utilizaremos combinaciones de 4 variables en lugar de las 5 utilizadas en el modelo anterior. Una de estas combinaciones será la de at_bats, hits, homeruns, y bat_avg, para el cuál el código utilizado será el siguiente:

```
mul5 <- lm(runs ~ at_bats + hits + homeruns + bat_avg, data = beis)
summary(mul5)
vif(mul5)
```

Del cuál obtenemos el siguiente modelo:

```
> summary(mul5)

Call:
lm(formula = runs ~ at_bats + hits + homeruns + bat_avg, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-41.559 -27.816  -8.172  24.474  57.496 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2387.7878  4025.4808   0.593    0.558    
at_bats        -0.5125     0.7292  -0.703    0.489    
hits            1.9019     2.8330   0.671    0.508    
homeruns        1.2210     0.1864   6.551 7.31e-07 ***
bat_avg     -6778.6913 15727.6216  -0.431    0.670    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 31.48 on 25 degrees of freedom
Multiple R-squared:  0.8744,	Adjusted R-squared:  0.8544 
F-statistic: 43.53 on 4 and 25 DF,  p-value: 6.484e-11
```

Y el siguiente VIF:

```
> vif(mul5)
    at_bats        hits    homeruns     bat_avg 
  99.306739 1781.464799    1.287672 1175.904585 
```

Usando el código mencionado anteriormente, al aplicarlo en las combinaciones restantes, da los siguientes resultados:

hits, homeruns, bat_avg y wins:

```
> summary(mul1)

Call:
lm(formula = runs ~ hits + homeruns + bat_avg + wins, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-50.070 -25.529  -4.479  25.722  59.339 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -396.5585   182.6700  -2.171 0.039638 *  
hits           0.1748     0.4790   0.365 0.718165    
homeruns       1.0461     0.2419   4.325 0.000214 ***
bat_avg     2420.0292  3279.2940   0.738 0.467402    
wins           0.8419     0.7539   1.117 0.274727    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 31.02 on 25 degrees of freedom
Multiple R-squared:  0.8781,	Adjusted R-squared:  0.8585 
F-statistic:    45 on 4 and 25 DF,  p-value: 4.523e-11

> vif(mul1)
     hits  homeruns   bat_avg      wins 
52.425167  2.232865 52.632062  2.231748 
```

at_bats, homeruns, bat_avg y wins:

```
> summary(mul2)

Call:
lm(formula = runs ~ at_bats + homeruns + bat_avg + wins, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-49.219 -25.890  -3.916  25.381  58.592 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -571.6802   557.8950  -1.025 0.315314    
at_bats        0.0284     0.1228   0.231 0.819027    
homeruns       1.0569     0.2416   4.375 0.000188 ***
bat_avg     3466.5835   780.3316   4.442 0.000158 ***
wins           0.7945     0.7509   1.058 0.300166    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 31.07 on 25 degrees of freedom
Multiple R-squared:  0.8777,	Adjusted R-squared:  0.8581 
F-statistic: 44.84 on 4 and 25 DF,  p-value: 4.705e-11

> vif(mul2)
 at_bats homeruns  bat_avg     wins 
2.890433 2.219920 2.970758 2.207328 
```

at_bats, hits, bat_avg y wins:

```
> summary(mul3)

Call:
lm(formula = runs ~ at_bats + hits + bat_avg + wins, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-85.528 -25.673  -6.174  23.821 100.398 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.057e+03  5.203e+03   0.587 0.562150    
at_bats     -6.240e-01  9.422e-01  -0.662 0.513845    
hits         3.231e+00  3.670e+00   0.880 0.387071    
bat_avg     -1.455e+04  2.042e+04  -0.713 0.482726    
wins         2.974e+00  7.505e-01   3.962 0.000546 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 40.66 on 25 degrees of freedom
Multiple R-squared:  0.7905,	Adjusted R-squared:  0.757 
F-statistic: 23.58 on 4 and 25 DF,  p-value: 3.563e-08

> vif(mul3)
    at_bats        hits     bat_avg        wins 
  99.346910 1791.455768 1187.479351    1.287549 
```

at_bats, hits, homeruns y wins:

```
> summary(mul4)

Call:
lm(formula = runs ~ at_bats + hits + homeruns + wins, data = beis)

Residuals:
    Min      1Q  Median      3Q     Max 
-48.851 -25.503  -5.091  24.989  58.735 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 316.7807   688.5595   0.460 0.649447    
at_bats      -0.1327     0.1502  -0.883 0.385535    
hits          0.6279     0.1394   4.505 0.000135 ***
homeruns      1.0526     0.2401   4.385 0.000183 ***
wins          0.8051     0.7447   1.081 0.289968    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 30.88 on 25 degrees of freedom
Multiple R-squared:  0.8792,	Adjusted R-squared:  0.8598 
F-statistic: 45.47 on 4 and 25 DF,  p-value: 4.038e-11

> vif(mul4)
 at_bats     hits homeruns     wins 
4.377185 4.481141 2.219620 2.197774 
```

Y al hacer la tabla con los valores pedidos, obtenemos esto:

Variables utilizadas en el modelo de regresión lineal múltiple	p-value del modelo	R^2 ajustado	VIF de esas variables
hits, homeruns, bat_avg, wins	4.523e-11	0.8585	52.425167 2.232865 52.632062 2.231748
at_bats, homeruns, bat_avg, wins	4.705e-11	0.8581	2.890433 2.219920 2.970758 2.207328
at_bats, hits, bat_avg, wins	3.563e-08	0.757	99.346910 1791.455768 1187.479351 1.287549
at_bats, hits, homeruns, wins	4.038e-11	0.8598	4.377185 4.481141 2.219620 2.197774
at_bats, hits, homeruns, bat_avg	6.484e-11	0.8544	99.306739 1781.464799 1.287672 1175.904585

Pregunta 15: ¿Considera que alguno de los modelos de cuatro variables de la tabla anterior predice mejor la variable de respuesta que el modelo de cinco variables (analizados en la pregunta 11)? ¿Cuál considera que es el mejor? Debe tomar en cuante que el modelo sea estadísticamente significativo y que no haya multicolinealidad entre las variables.

En el modelo de cinco variables obtuvimos un Adjusted R-squared de 85.6%, mientras que en estos modelos de cuatro variables, el más alto fue de 85.98% por la combinación de at_bats, hits, homeruns y wins, por lo que aparte de ser este último el mejor de todos los modelos de cuatro variables realizados, también es mejor que el modelo de cinco variables realizado.