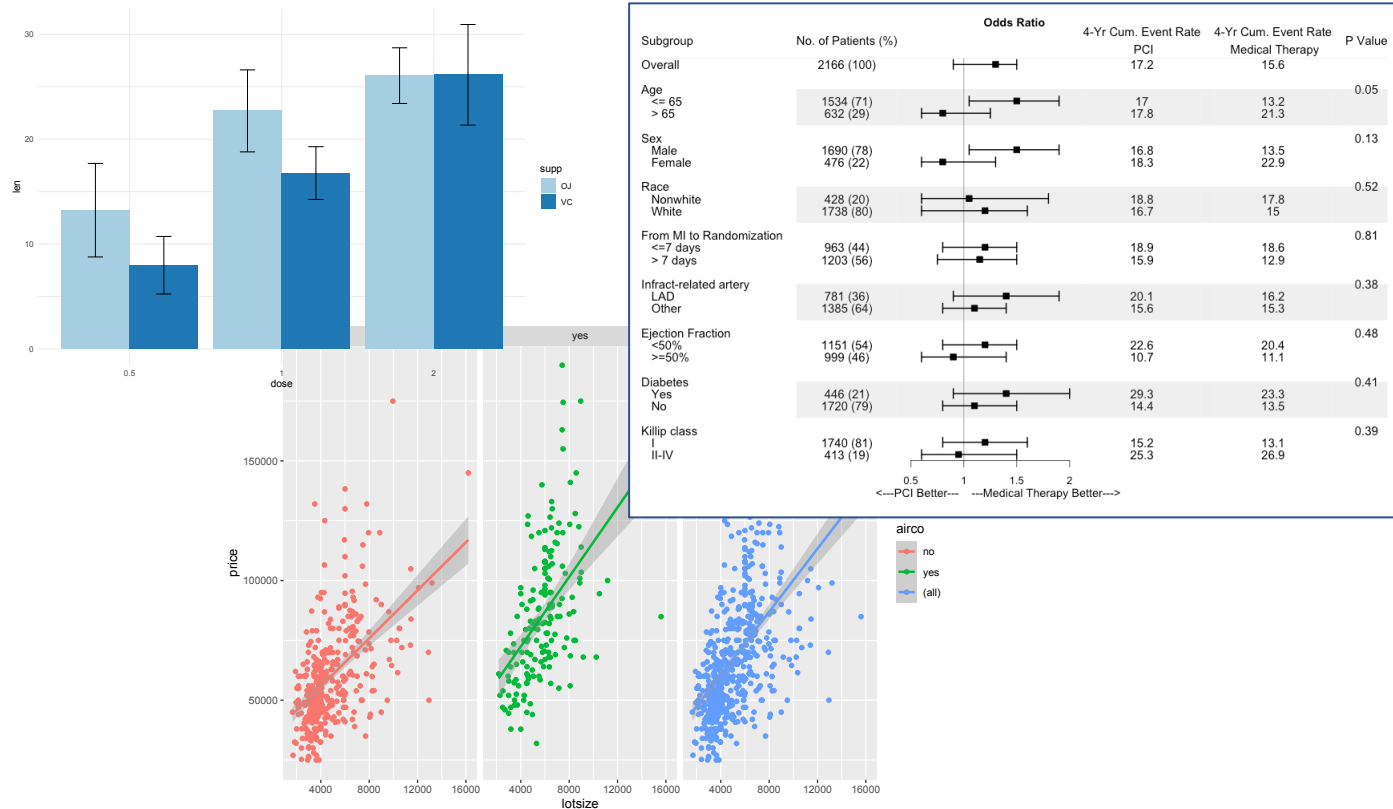


Análisis Estadístico con



6. Regresión y Correlación

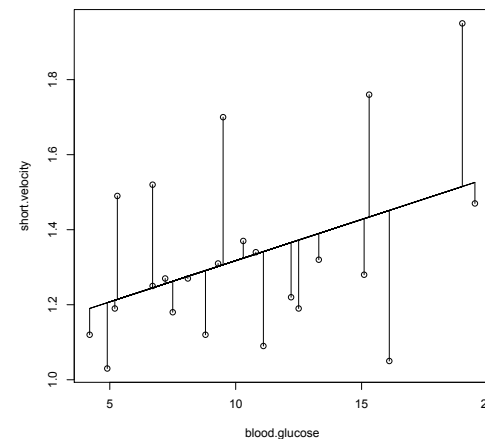
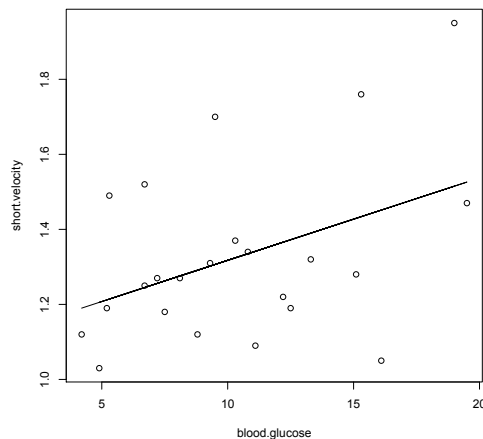
Índice

- 6.1. Regresión Lineal Simple
- 6.2. Resíduos y Valores ajustados
- 6.3. Predicción y bandas de confianza
- 6.4. Correlación

6. Regresión y Correlación

6.1. Regresión Lineal Simple

- Describir la relación entre dos variables. Asociación.
- $Y = a + bX + E$
- Donde $E \sim N(0, s^2)$
- Método de estimación de a , b y s^2 : mínimos cuadrados.
- Encontrar los valores de a y b que minimicen la suma de los residuos al cuadrado.



6. Regresión y Correlación

6.1. Regresión Lineal Simple

- **lm()**
- **summary(lm())**
- inferencia parámetros
(IC, test de hipótesis)

	blood.glucose	short.velocity
1	15.3	1.76
2	10.8	1.34
3	8.1	1.27
4	19.5	1.47
5	7.2	1.27
6	5.3	1.49
7	9.3	1.31
8	11.1	1.09
9	7.5	1.18
10	12.2	1.22
11	6.7	1.25
12	5.2	1.19
13	19.0	1.95
14	15.1	1.28
15	6.7	1.52
16	8.6	NA
17	4.2	1.12
18	10.3	1.37
19	12.5	1.19
20	16.1	1.05
21	13.3	1.32
22	4.9	1.03
23	8.8	1.12
24	9.5	1.70

```
> lm(short.velocity~blood.glucose)
```

```
Call:
lm(formula = short.velocity ~ blood.glucose)
```

```
Coefficients:
(Intercept)  blood.glucose
 1.09781      0.02196
```

```
> summary(lm(short.velocity~blood.glucose))
```

```
Call:
lm(formula = short.velocity ~ blood.glucose)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-0.40141 -0.14760 -0.02202  0.03001  0.43490
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.09781    0.11748   9.345 6.26e-09 ***
blood.glucose  0.02196    0.01045   2.101  0.0479 *
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.2167 on 21 degrees of freedom
(1 observation deleted due to missingness)
```

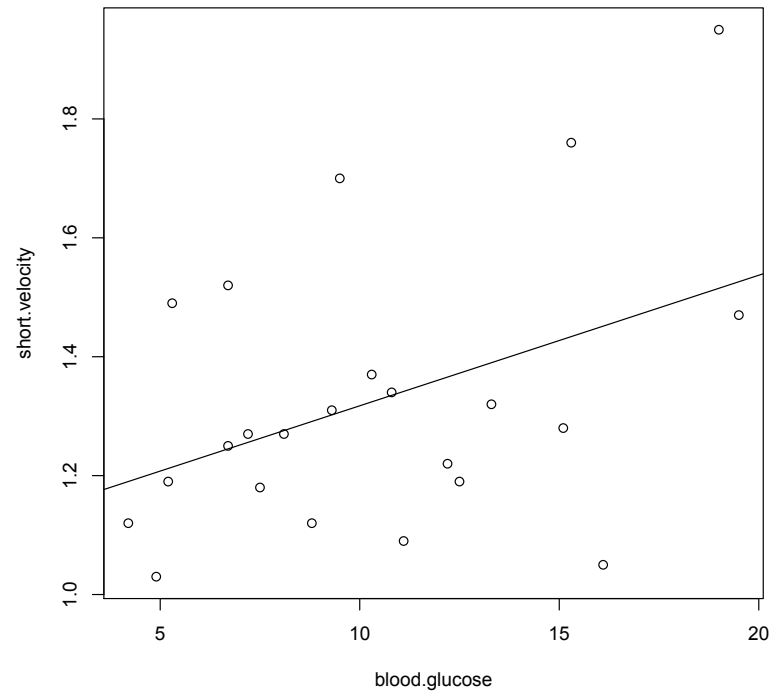
```
Multiple R-squared:  0.1737,    Adjusted R-squared:  0.1343
F-statistic: 4.414 on 1 and 21 DF,  p-value: 0.0479
```

6. Regresión y Correlación

6.1. Regresión Lineal Simple

- **plot**(x,y)
- **abline**(lm())

```
> plot(blood.glucose,short.velocity)  
> abline(lm(short.velocity~blood.glucose))
```



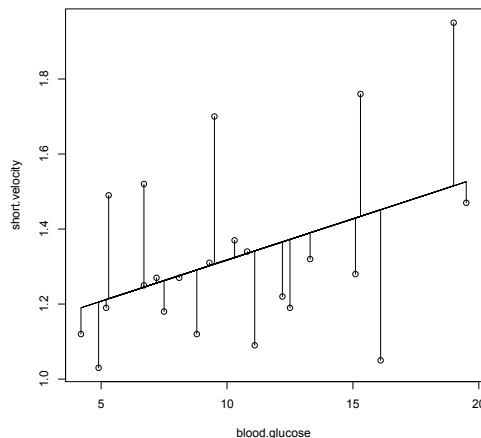
6. Regresión y Correlación

6.2. Resíduos y Valores ajustados

- **fitted()** # Valores que esperarías en Y dado los valores de x de la muestra siguiendo el modelo de regresión lineal con los parámetros estimados por mínimos cuadrados
- **resid()** # Resíduos= diferencia entre los valores de y de la muestra y los valores ajustados.

```

lm.velo<-lm(short.velocity~blood.glucose)
fitted(lm.velo)
resid(lm.velo)
  
```



	blood.glucose	short.velocity	fitted	resid
1	15.3	1.76	1.433841	0.326158532
2	10.8	1.34	1.335010	0.004989882
3	8.1	1.27	1.275711	-0.005711308
4	19.5	1.47	1.526084	-0.056084062
5	7.2	1.27	1.255945	0.014054962
6	5.3	1.49	1.214216	0.275783754
7	9.3	1.31	1.302066	0.007933665
8	11.1	1.09	1.341599	-0.251598875
9	7.5	1.18	1.262534	-0.082533795
10	12.2	1.22	1.365758	-0.145757649
11	6.7	1.25	1.244964	0.005036223
12	5.2	1.19	1.212020	-0.022019994
13	19.0	1.95	1.515103	0.434897199
14	15.1	1.28	1.429449	-0.149448964
15	6.7	1.52	1.244964	0.275036223
16	8.6	NA	NA	NA
17	4.2	1.12	1.190057	-0.070057471
18	10.3	1.37	1.324029	0.045971143
19	12.5	1.19	1.372346	-0.182346406
20	16.1	1.05	1.451411	-0.401411486
21	13.3	1.32	1.389916	-0.069916424
22	4.9	1.03	1.205431	-0.175431237
23	8.8	1.12	1.291085	-0.171085074
24	9.5	1.70	1.306459	0.393541161

6. Regresión y Correlación

6.3. Predicción y bandas de confianza

- **predict**((lm()))==fitted()
- **predict**((lm()),interval="confidence")
- lm.velo<-lm(short.velocity~blood.glucose)
- predict(lm.velo,interval="confidence")

```

predict(lm.velo,int="c")
      fit      lwr      upr
1.433841 1.291371 1.576312
1.335010 1.240589 1.429431
1.275711 1.169536 1.381887
1.526084 1.306561 1.745607
1.255945 1.139367 1.372523
1.214216 1.069315 1.359118
1.302066 1.205244 1.398889
1.341599 1.246317 1.436881
1.262534 1.149694 1.375374
1.365758 1.263750 1.467765
1.244964 1.121641 1.368287
1.212020 1.065457 1.358583
1.515103 1.305352 1.724854
1.429449 1.290217 1.568681
1.244964 1.121641 1.368287
      NA      NA      NA
1.190057 1.026217 1.353898
1.324029 1.230050 1.418008
1.372346 1.267629 1.477064
1.451411 1.295446 1.607377
1.389916 1.276444 1.503389
1.205431 1.053805 1.357057
1.291085 1.191084 1.391086
1.306459 1.210592 1.402326
  
```

6. Regresión y Correlación

6.4. Correlación

- Coeficiente de Correlación: parámetro para medir la asociación lineal entre dos variables aleatorias
- Valores: entre -1 y +1
- El 0 indica no correlación, y el signo el sentido de la asociación.
- Tipos:
 - **Pearson** (r): `cor(x,y)` # cuidado con missing (`cor(x,y,use="complete.obs")`)
`sqrt(summary(lm())$r.squared)`
 - **Coeficiente de determinación** (R^2): `summary(lm())$r.squared`
 - **Spearman** (ρ): `cor(x,y,method="spearman")`
 - **Kendall** (T): `cor(x,y,method="kendall")`
- **cor.test()**

6. Regresión y Correlación

6.4. Correlación

```
cor(blood.glucose,short.velocity, use="complete.obs")
```

```
cor.test(blood.glucose,short.velocity,method="spearman")
```

```
cor.test(blood.glucose,short.velocity,method="kendall")
```

```
> cor(blood.glucose,short.velocity, use="complete.obs")
[1] 0.4167546
> cor.test(blood.glucose,short.velocity,method="spearman")

Spearman's rank correlation rho

data: blood.glucose and short.velocity
S = 1380.4, p-value = 0.1392
alternative hypothesis: true rho is not equal to 0
sample estimates:
rho
0.318002

Warning message:
In cor.test.default(blood.glucose, short.velocity, method = "spearman") :
  Cannot compute exact p-value with ties
> cor.test(blood.glucose,short.velocity,method="kendall")

Kendall's rank correlation tau

data: blood.glucose and short.velocity
z = 1.5604, p-value = 0.1187
alternative hypothesis: true tau is not equal to 0
sample estimates:
tau
0.2350616

Warning message:
In cor.test.default(blood.glucose, short.velocity, method = "kendall") :
  Cannot compute exact p-value with ties
>
```

EJERCICIOS

“ejercicios.6.regresion.y.correlación.R”