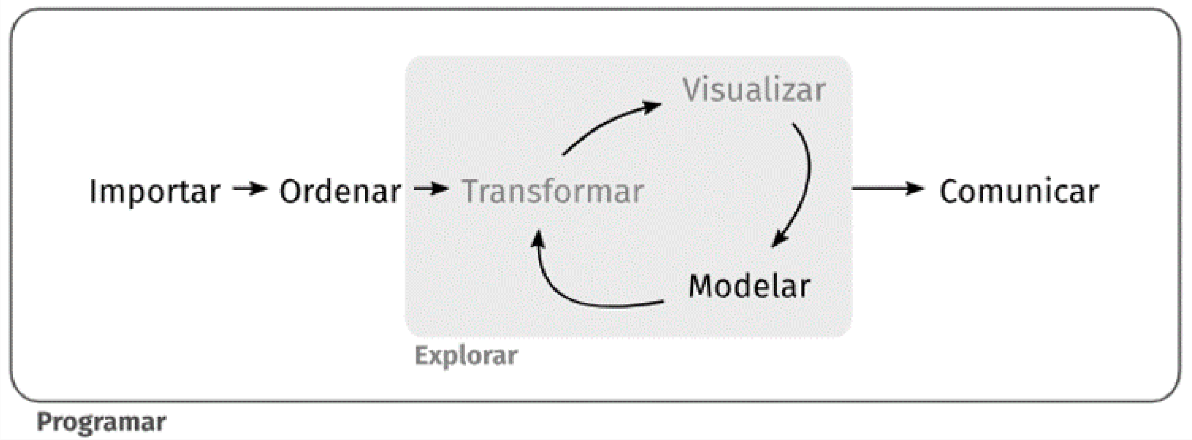
Ciencia de Datos para Políticas Públicas

Módulo 2 - Clase 6: Logit y otros

Pablo Aguirre Hormann
20/07/2021

¿Qué veremos hoy?

- Visualización de datos
- Manejo de datos
- Transformación de datos
- **Inferencia Estadística/Econometría**
  - Modelo logit
  - Aplicaciones



Modelos

Objetivo: representar la relación entre una variable dependiente Y y una o varias variables explicativas/independientes $X_1, X_2, \ldots, X_k$.

$$\begin{aligned}\hat{Y} &= E(Y|X)\\ &= \hat{f}\,(X)\end{aligned}$$

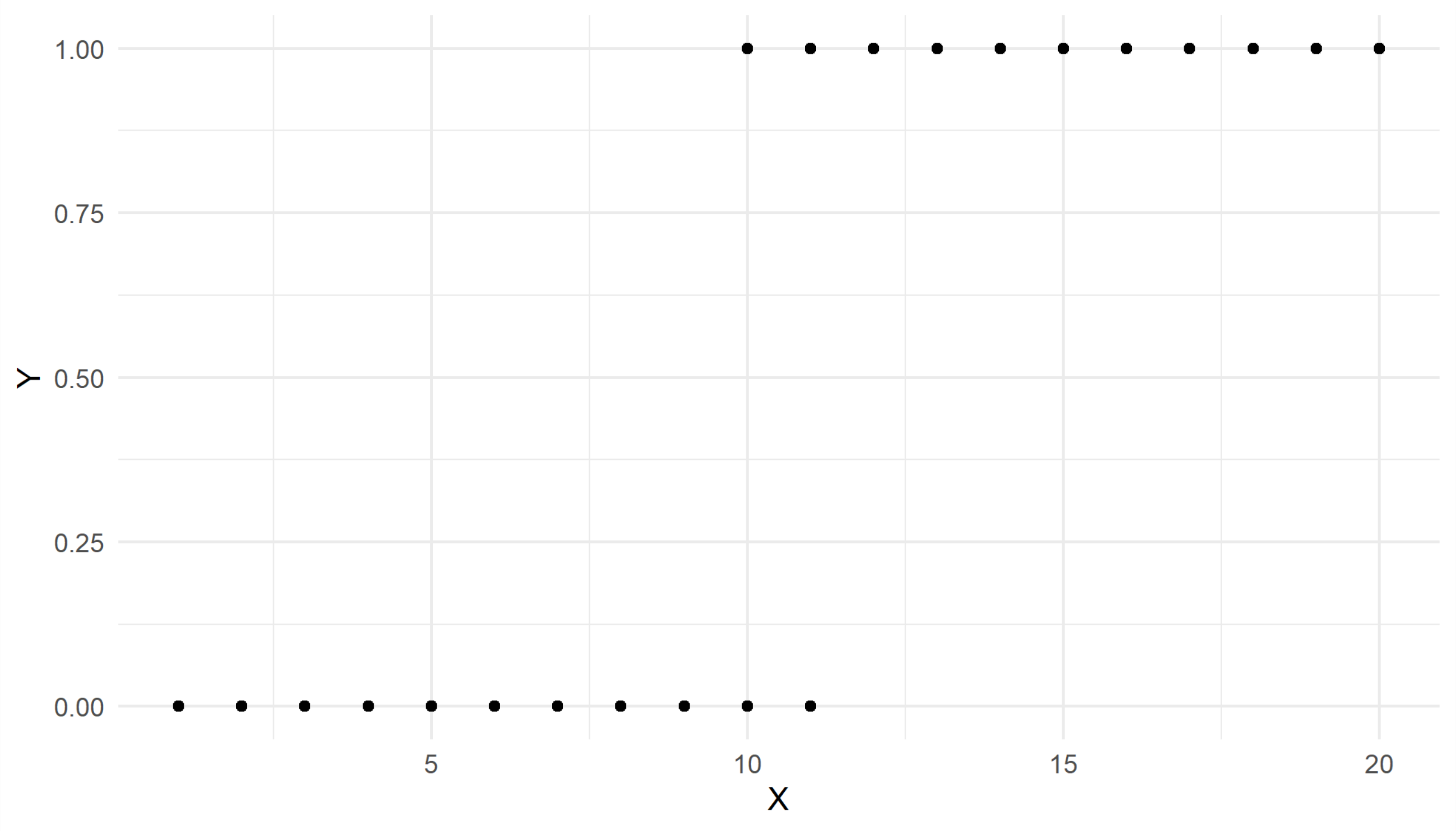
- Si Y es una variable *continua*: **regresión**
- Si Y es una variable *categórica*: **clasificación** (próxima clase)

Regresión Logística/Clasificación

Variable dependiente binaria

- Hasta ahora consideramos una variable dependiente Y continua (Resultados de prueba)
- Pero también podemos tener casos en que Y es una variable categórica/binaria (1 o 0)
    - Otorgamiento de subsidio (sí/no)
    - Participación en el mercado laboral (sí/no)
- Esto conlleva algunos desafíos extra a los que hemos visto hasta ahora

Variable dependiente binaria



No existe dispersión en el eje Y

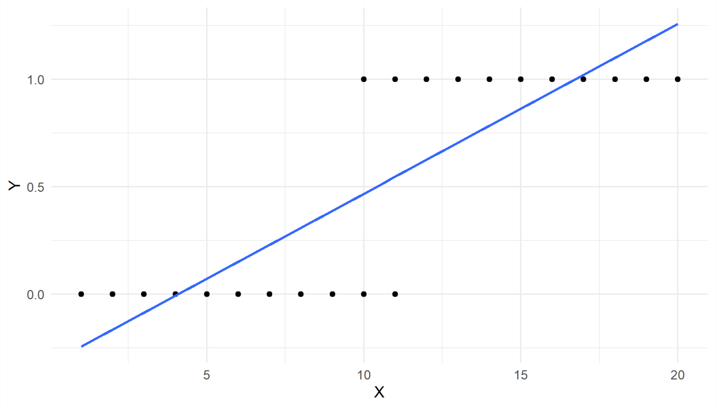
¿Qué ocurre si modelamos esto al igual que una regresión con Y continua?

Modelo de probabilidad lineal

$$\hat{Y} = P(Y = 1|X) = \hat{\beta}_0 + \hat{\beta}_1 X$$

```
lm(Y~X, data = datos_logit)
```

```
##
## Call:
## lm(formula = Y ~ X, data = datos_logit)
##
## Coefficients:
## (Intercept)            X
##     -0.32360      0.07912
```



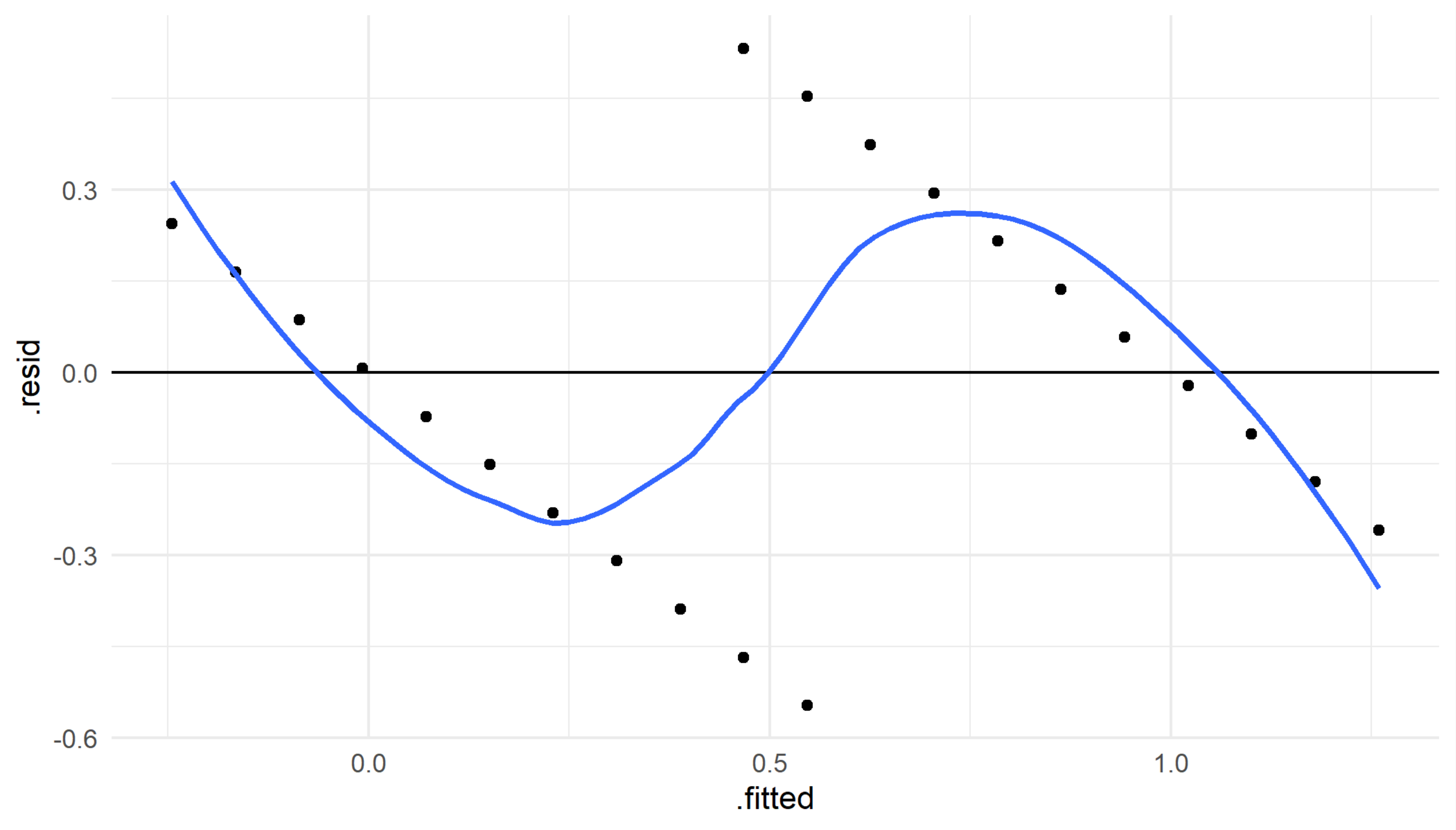
El modelo de probabilidad lineal tiene la ventaja de que la interpretación es directa, *el aumento en una unidad de* X *esta asociado, en promedio, con un aumento de 7.9% en* Y

Pero modelo permite valores ajustados menores a 0 y superiores a 1.

¿Cómo interpretamos, por ejemplo, $\hat{Y} = 1.2$?

Residuales

Otro problema: los residuales claramente muestran que algo anda mal.



Debemos buscar una forma de limitar los valores de Y: $P(Y = 1|X) = F(\hat{\beta}_0 + \hat{\beta}_1 X)$

Modelo logit

El modelo logit (o logístico) nos permite limitar los valores de Y entre 0 y 1 usando como función auxiliar $F = \frac{exp(z)}{1+exp(z)}$ con $z = \hat{\beta}_0 + \hat{\beta}_1 X$.

$$P(Y = 1|X) = \frac{e^{(\hat{\beta}_0 + \hat{\beta}_0 X)}}{1 + e^{(\hat{\beta}_0 + \hat{\beta}_0 X)}}$$

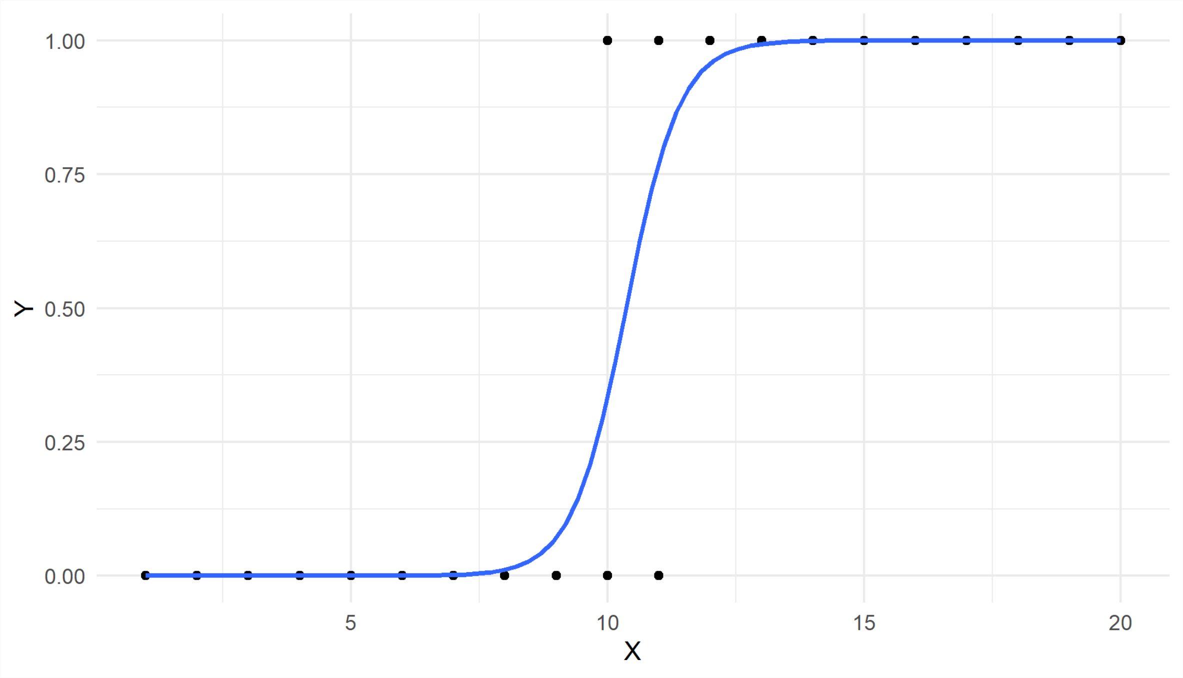
El proceso de estimación es algo distinto a lo que vimos para regresiones hasta el momento. En este caso se hace por algo llamado **máxima verosimilitud** (no entraremos en detalles).

Pero en R...

```
modelo_logit <- glm(Y ~ X, family = "binomial", data = datos_logit)
```

Noten que usamos `glm` ahora y no `lm`.

¿Cómo se ve esto?




```
##
## Call:  glm(formula = Y ~ X, family = "binomial", data = datos_logit)
##
## Coefficients:
## (Intercept)            X
##     -19.646        1.896
##
## Degrees of Freedom: 199 Total (i.e. Null);  198 Residual
## Null Deviance:        277.3
## Residual Deviance: 61.44     AIC: 65.44
```

$$P(Y=1|X)=\frac{e^{(-19.6+1.9X)}}{1+e^{(-19.6+1.9X)}}$$

Con datos reales

Datos laborales

```r
(datos_trabajo <- read_xlsx("../datos/datos_trabajo.xlsx"))
```

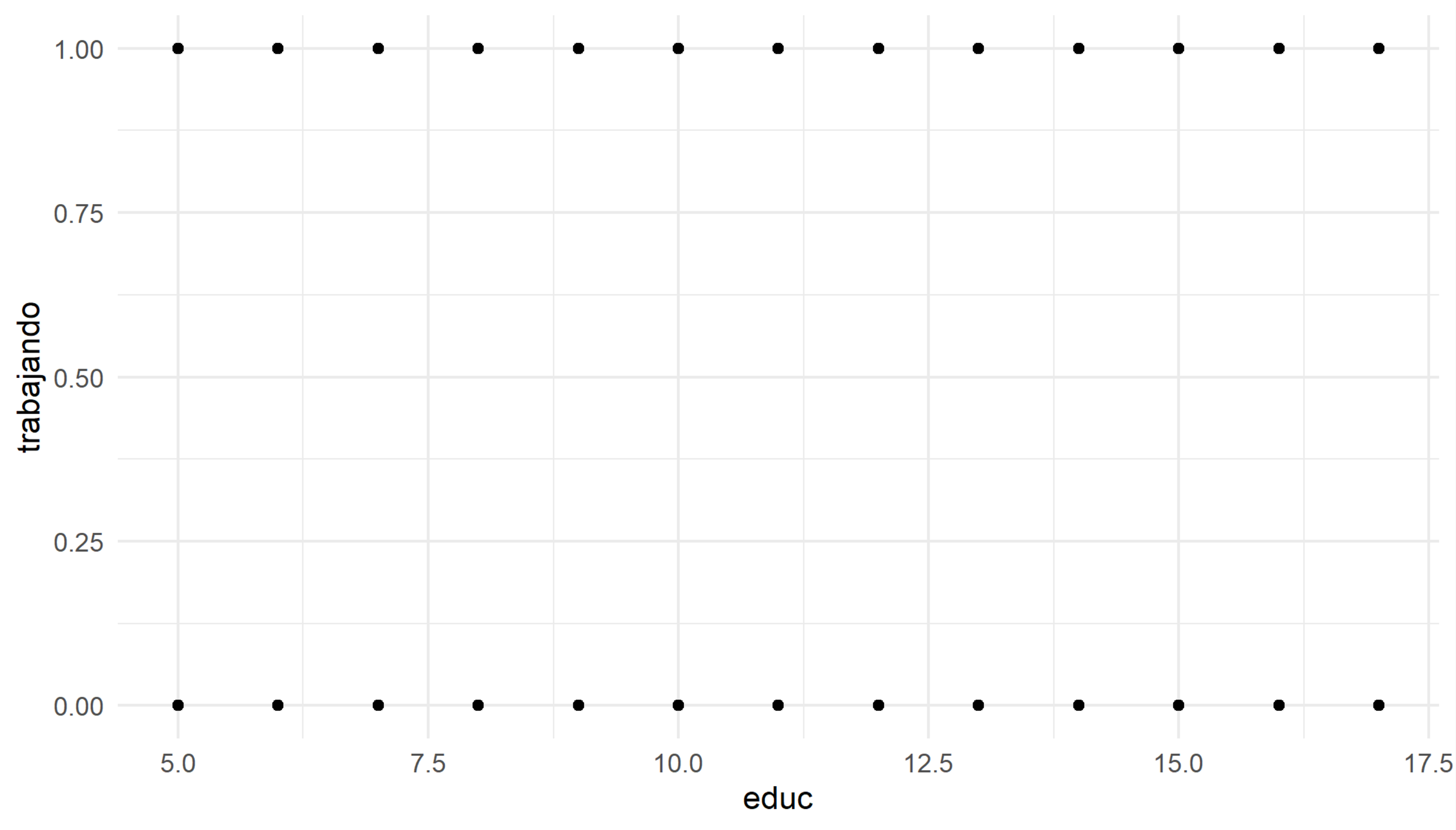
```
## # A tibble: 687 x 4
##    trabajando  educ exper exper_cuad
##         <dbl> <dbl> <dbl>      <dbl>
##  1          0    11     1          1
##  2          1    13     4         16
##  3          1    11     4         16
##  4          1    12    19        361
##  5          1     8     2          4
##  6          0    17    10        100
##  7          1     6    14        196
##  8          0    17     1          1
##  9          1     8    25        625
## 10          1    10    11        121
## # ... with 677 more rows
```

- `trabajando` es una variable categórica que toma el valor **1** cuando una persona/observación se encuentra trabajando y **0** en caso contrario.

- `educ` y `exper` son variables numéricas representando años de educación y de experiencia laboral, respectivamente.

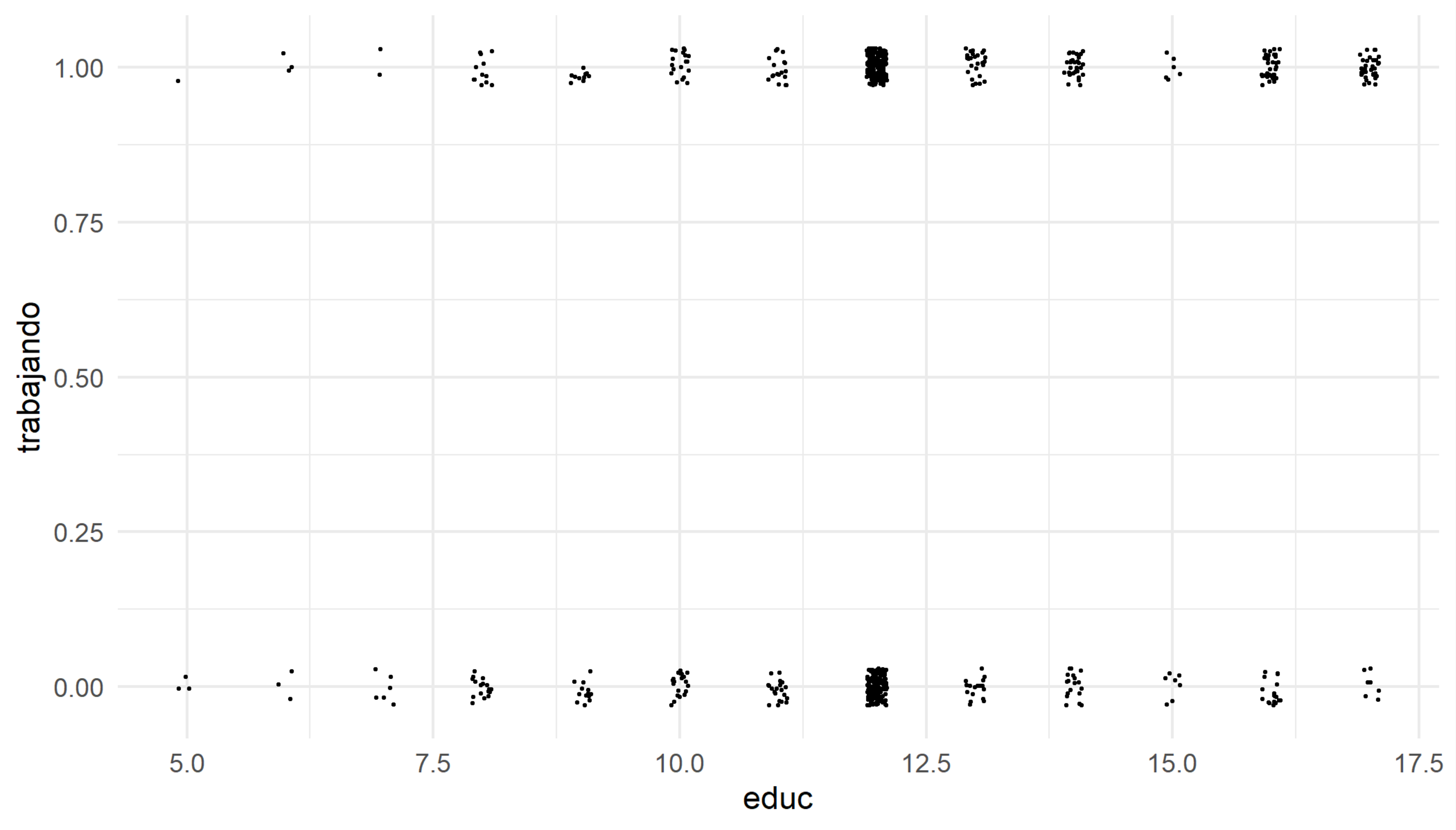
Visualicemos los datos

```
datos_trabajo %>%
  ggplot(aes(x = educ, y = trabajando)) +
  geom_point() +
  theme_minimal()
```



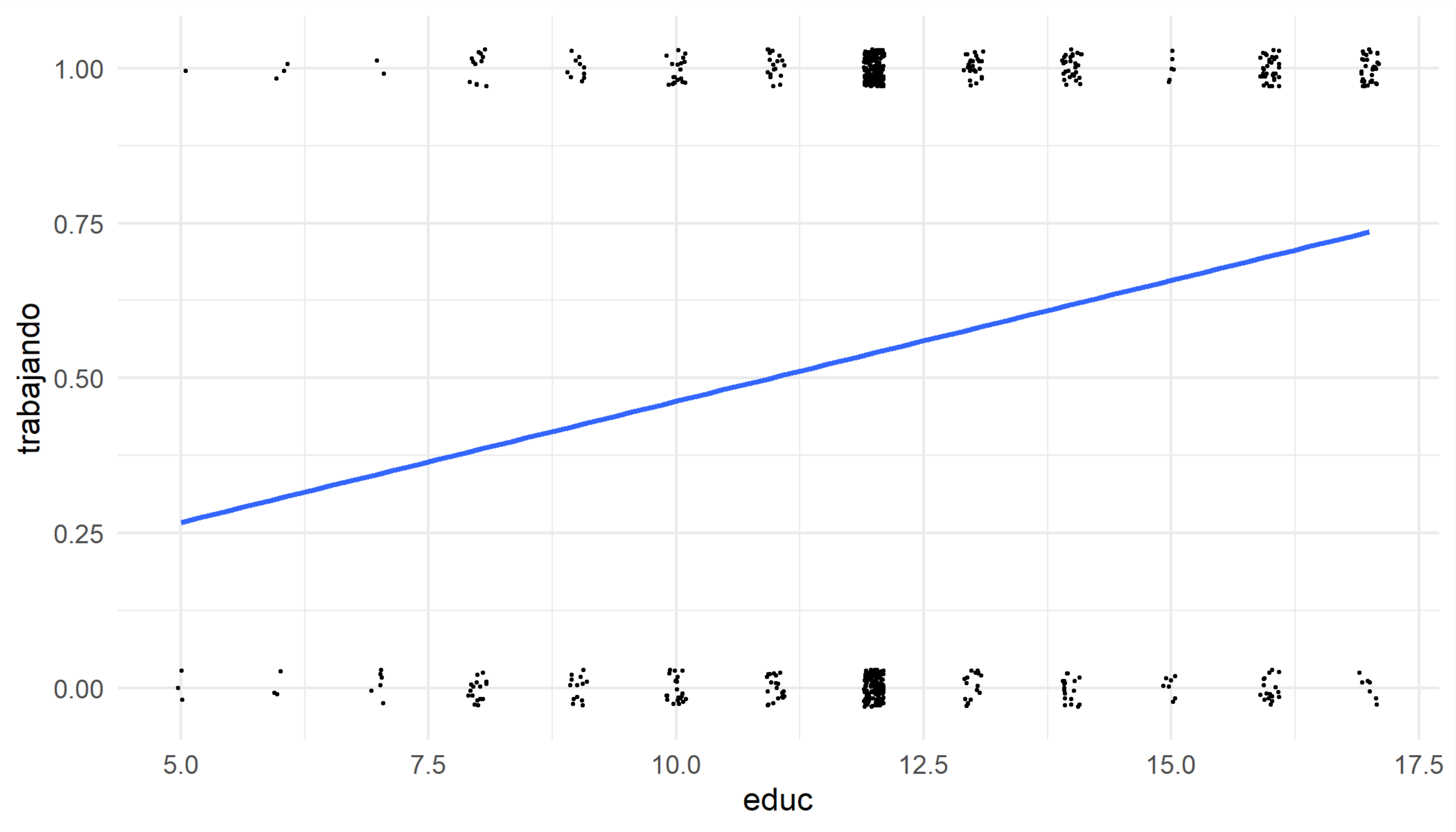
Un pequeño ajuste

```
datos_trabajo %>%
  ggplot(aes(x = educ, y = trabajando)) +
  geom_jitter(width = 0.1, height = 0.03, size = 0.3) +
  theme_minimal()
```



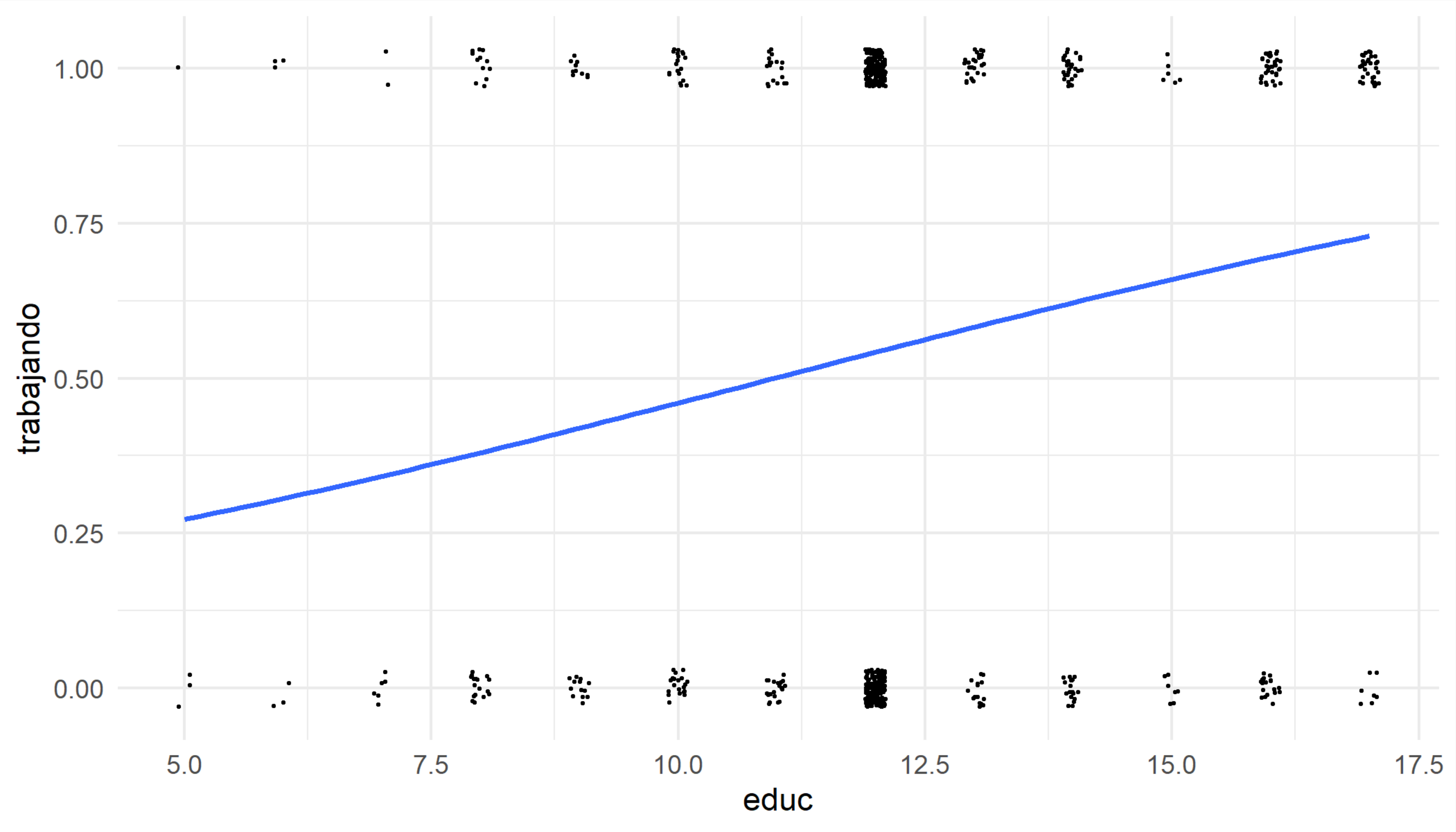
Modelo de probabilidad lineal

```
datos_trabajo %>%
  ggplot(aes(x = educ, y = trabajando)) +
  geom_jitter(width = 0.1, height = 0.03, size = 0.3) +
  geom_smooth(method = "lm", se = FALSE) +
  theme_minimal()
```



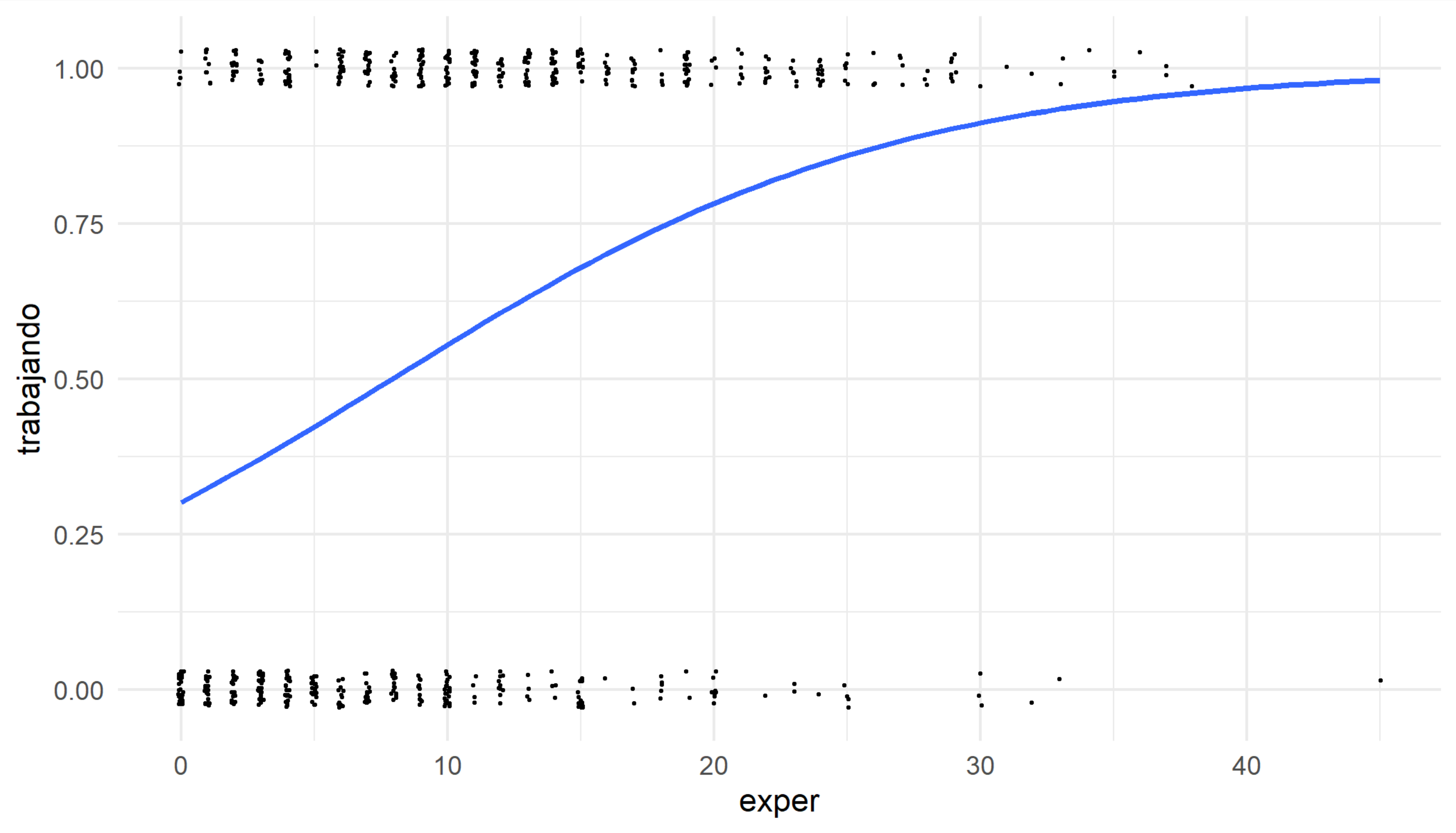
Modelo logit

```
datos_trabajo %>%
  ggplot(aes(x = educ, y = trabajando)) +
  geom_jitter(width = 0.1, height = 0.03, size = 0.3) +
  geom_smooth(method = "glm", se = FALSE, method.args = list(family = "binomial")) +
  theme_minimal()
```



Modelo logit con otra variable

```
datos_trabajo %>%
  ggplot(aes(x = exper, y = trabajando)) +
  geom_jitter(width = 0.1, height = 0.03, size = 0.3) +
  geom_smooth(method = "glm", se = FALSE, method.args = list(family = "binomial")) +
  theme_minimal()
```



Estimemos un modelo logit

```
modelo_logit_trabajo <- glm(trabajando ~ educ,
                            family = "binomial",
                            data = datos_trabajo)
tidy(modelo_logit_trabajo)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic    p.value
##   <chr>          <dbl>     <dbl>     <dbl>      <dbl>
## 1 (Intercept)   -1.80     0.438      -4.12 0.0000383
## 2 educ           0.164    0.0354      4.65 0.00000332
```

$$P(trabajando = 1|educ) = \frac{e^{(-1.8+0.16educ)}}{1+e^{(-1.8+0.16*educ)}}$$

¿Cómo interpretamos esto?

Un poco de algebra

$$P(trabajando = 1|educ) = p = \frac{e^{(-1.8+0.16*educ)}}{1 + e^{(-1.8+0.16*educ)}}$$

$$\frac{1}{p} = \frac{1 + e^{(-1.8+0.16*educ)}}{e^{(-1.8+0.16*educ)}}$$

$$\frac{1}{p} = 1 + \frac{1}{e^{(-1.8+0.16*educ)}}$$

$$\frac{1-p}{p} = \frac{1}{e^{(-1.8+0.16*educ)}}$$

$$\frac{p}{1-p} = e^{(-1.8+0.16*educ)}$$

$$log(\frac{p}{1-p}) = -1.8 + 0.16 * educ$$

$$log\left(\frac{P(trabajando = 1|educ)}{P(trabajando = 0|educ)}\right) = -1.8 + 0.16 * educ$$

Interpretación

$$log\left(\frac{P(trabajando=1|educ)}{P(trabajando=0|educ)}\right) = -1.8 + 0.16 * educ$$

El aumento en una unidad de `educ` se asocia con un incremento promedio de 0.16 en el ***log-odds*** de `trabajando`.

El efecto depende del "lugar de la curva" donde estemos.

```
nuevos_datos <- data.frame("educ" = c(5, 7, 9, 11, 13, 15, 17))
predict(modelo_logit_trabajo,
        newdata = nuevos_datos,
        type = "response") %>% round(4)
```

```
##      1      2      3      4      5      6      7
## 0.2724 0.3422 0.4195 0.5010 0.5825 0.6597 0.7292
```

¿Cómo evaluamos este modelo?

Pseudo R^2

Logit es un ejemplo de modelos de regresión no lineal y es importante destacar que en estos casos una métrica como el R^2 no tiene sentido ya que sus supuestos son para modelos lineales.

Una alternatva es utilizar una métrica conocida como el *pseudo*- R^2.

$$pseudo\ R^2 = 1 - \frac{ln(f_{full}^{max})}{ln(f_{nulo}^{max})} = 1 - \frac{devianza}{devianza\ nula}$$

```
glance(modelo_logit_trabajo)
```

```
## # A tibble: 1 x 8
##   null.deviance df.null logLik   AIC   BIC deviance df.residual  nobs
##           <dbl>   <int>  <dbl> <dbl> <dbl>    <dbl>       <int> <int>
## 1          945.     686  -461.  926.  935.     922.         685   687
```

```
1-(select(glance(modelo_logit_trabajo), deviance)/select(glance(modelo_logit_trabajo), null.deviance)) %>% pull()
```

```
## [1] 0.02437236
```

¿Cómo evaluamos este modelo?

```r
summary(modelo_logit_trabajo)
```

```
##
## Call:
## glm(formula = trabajando ~ educ, family = "binomial", data = datos_trabajo)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6165  -1.2498   0.8521   1.1067   1.6128
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.80473    0.43832  -4.117 3.83e-05 ***
## educ         0.16444    0.03536   4.650 3.32e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 945.03  on 686  degrees of freedom
## Residual deviance: 922.00  on 685  degrees of freedom
## AIC: 926
##
## Number of Fisher Scoring iterations: 4
```

¿Cómo evaluamos este modelo?

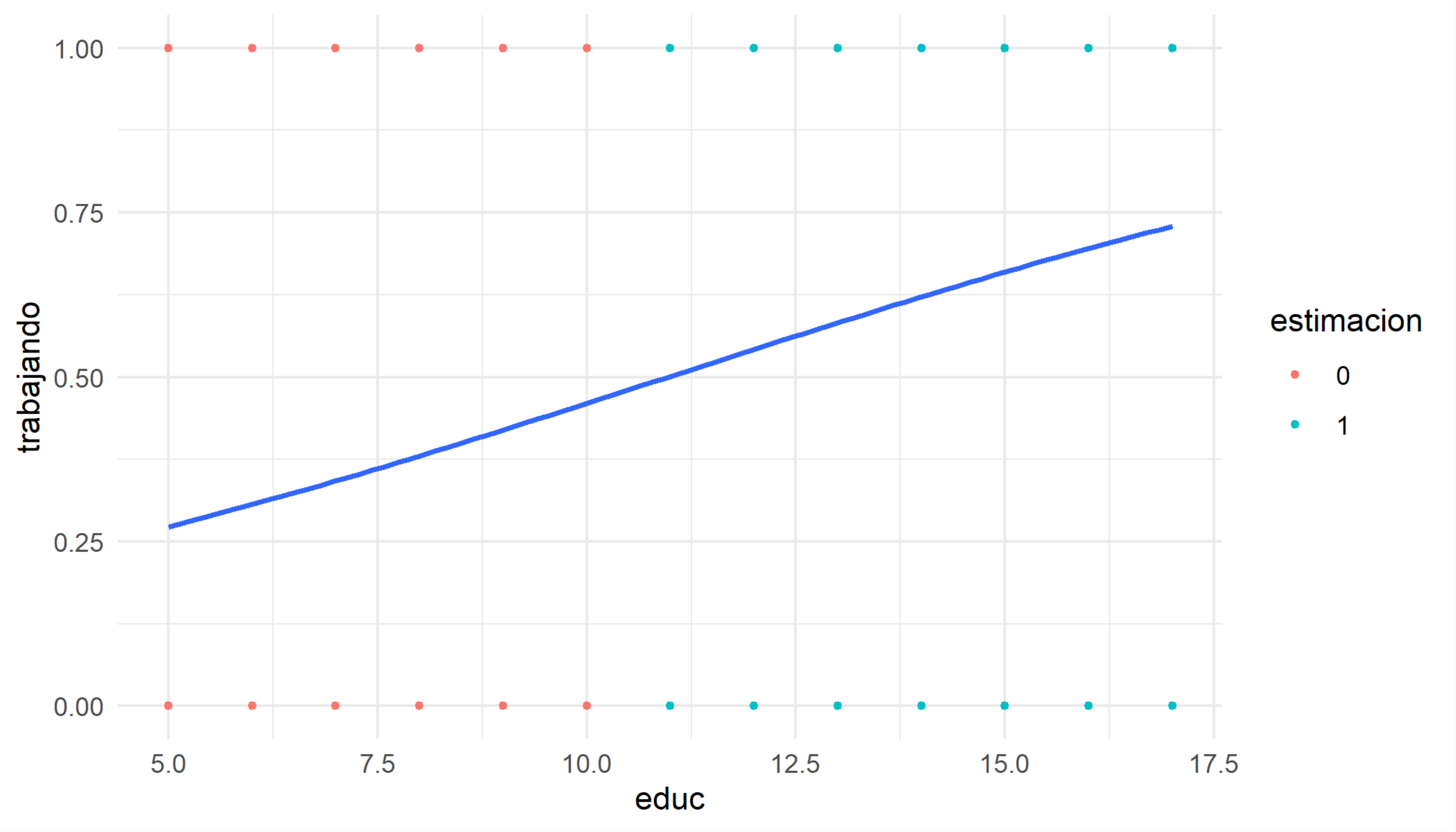
Otra forma de evaluar es convertir los *valores ajustados* (resultado del modelo) que corresponde a valores entre 0 y 1 (ver `.fitted`) en categorías que se puedan comparar con `trabajando` (0 o 1).

```
(estimacion_logit <- augment(modelo_logit_trabajo, type.predict = "response") %>%
  transmute(valor_real = trabajando,
            .fitted,
            valor_estimado = ifelse(.fitted >= 0.5, 1, 0),
            check = valor_real == valor_estimado))
```

```
## # A tibble: 687 x 4
##    valor_real .fitted valor_estimado check
##         <dbl>   <dbl>          <dbl> <lgl>
##  1          0   0.501              1 FALSE
##  2          1   0.582              1 TRUE
##  3          1   0.501              1 TRUE
##  4          1   0.542              1 TRUE
##  5          1   0.380              0 FALSE
##  6          0   0.729              1 FALSE
##  7          1   0.306              0 FALSE
##  8          0   0.729              1 FALSE
##  9          1   0.380              0 FALSE
## 10          1   0.460              0 FALSE
## # … with 677 more rows
```

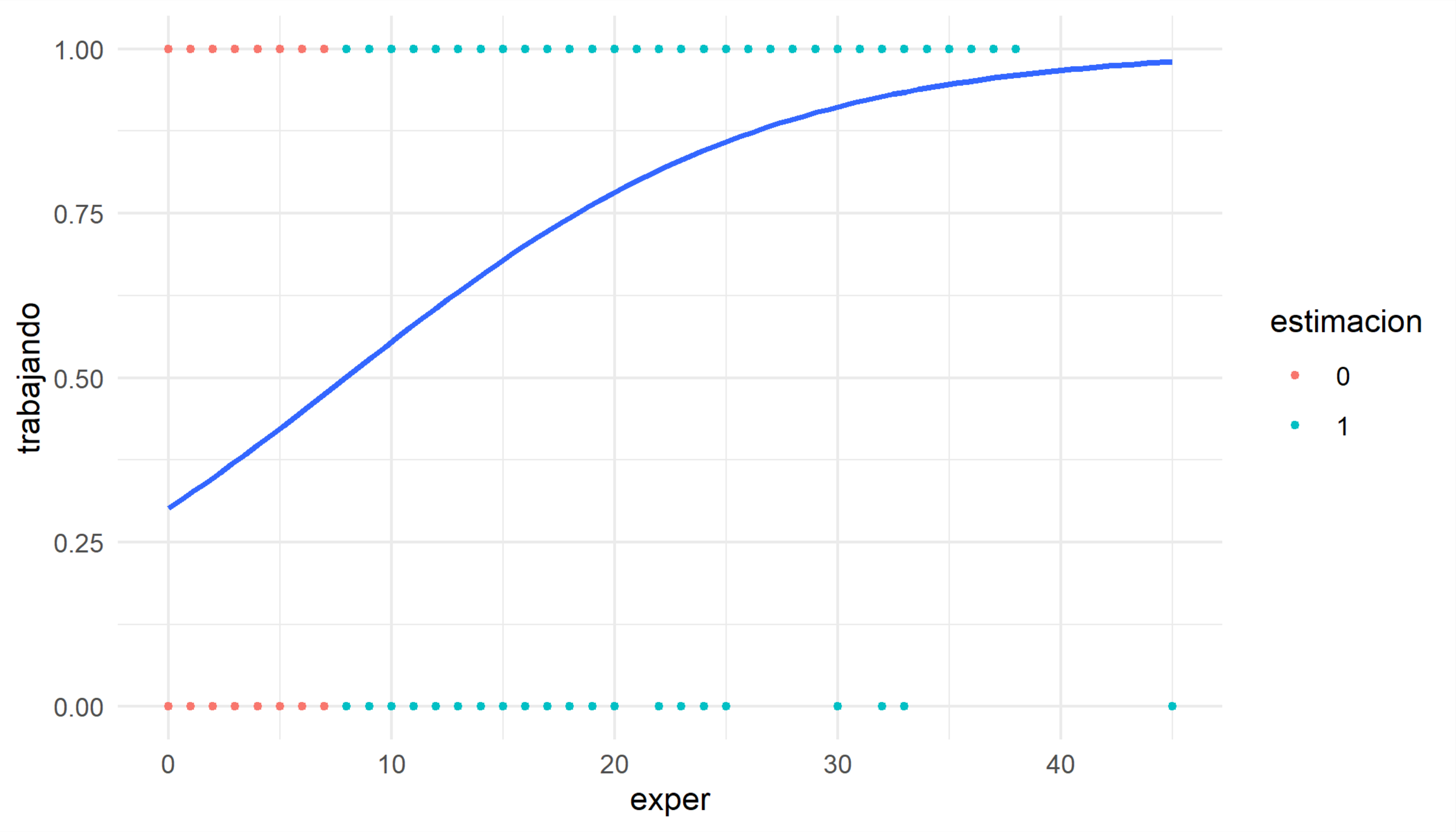
¿Qué significa esto?

Clasificación usando 0.5



¿Qué significa esto?

Clasificación usando 0.5 para otra variable X



Matriz de confusión

```
(matriz_confusion <- estimacion_logit %>%
  group_by(valor_real, valor_estimado) %>%
  summarise(n = n()) %>%
  pivot_wider(names_from = valor_estimado, values_from = n))
```

```
## # A tibble: 2 x 3
## # Groups:   valor_real [2]
##   valor_real   `0`   `1`
##        <dbl> <int> <int>
## 1          0    63   245
## 2          1    47   332
```

No pareciera ser un buen modelo

```
VP <- matriz_confusion[2,3]
FP <- matriz_confusion[1,3]
VN <- matriz_confusion[1,2]
FN <- matriz_confusion[2,2]
(tasa_VP <- VP/(VP+FN))
##           1
## 1 0.8759894
(tasa_FP <- FP/(FP+VN))
##           1
## 1 0.7954545
```

En general, queremos maximizar la tasa de Verdados Positivos y minimizar la tasa de Falsos Positivos.

Inferencia vs Predicción

Modelos para Inferencia/Explicación:

- Aprender y concluir algo sobre como se relacionan variables. Relaciones causales.
- Evitar sesgo
- Predicción *dentro de muestra*
- $\hat{f}$ / $\hat{\beta}$

Modelos para Predicción:

- Que la predicción esté lo más cerca posible del valor real
- Evitar sobreajuste al entrenar modelos
- Predicción *fuera de muestra*
- $\hat{Y}$

Aplicaciones

SMA - Preferencias de fiscalización

Situación: Elaboración de programas de fiscalización a partir de criterio experto (muy valioso) considerando información "objetiva" (denuncias, por ej.) e información "subjetiva" (percepciones).

Antecedente: ¿Cómo ordenar el proceso de manera de sistematizar al menos la información "objetiva".

- **Solución propuesta**: a través de ponderadores para distintos criterios, generar un ranking de establecimientos a fiscalizar.
- **Problema**: ¿por qué valorar un componente más que otro?

Propuesta actual: de alguna forma capturar la preferencia de funcionarios/as de la SMA a distintos criterios "objetivos". Con esas preferencia **estimar los "ponderadores"**.

Experimento de elección

- Basados en la teoría de utilidad aleatoria (McFadden 1973) que propone que la utilidad de un bien se descompone en un **componente observable** y otro **no observable** (error).
- Los componentes observables corresponden a **atributos ligados a las elecciones** que un individuo hace así como a **características del individuo** que hace la elección.
- Dados ciertos supuestos para la distribución del error, la **probabilidad de elegir una opción se puede expresar como una distribución logística**.

En la práctica

- 48 elecciones a casi 200 funcionarios/as de distintas áreas.
- **¿Qué establecimiento fiscalizarías?**
- Cada elección se traduce en elegir una opción A o una opción B.
- Cada elección depende de las características descritas y también de la persona que responde.
- Esto nos deja con una base de datos de casi 10.000 observaciones que podemos modelar.

Resultados

$$\widehat{Utilidad} = (0.32 * 1_{Aire}) + (0.34 * 1_{Ext.Ag}) + (0.36 * 1_{FyF}) + (0.09 * 1_{RuO}) + (0.47 * 1_{Ag\_Suelo}) + (0.83 * 1_{Con\_ImpSign}) + (0.81 * 1_{Con\_Den}) + (0.74 * 1_{Sin\_Fisc}) + (0.25 * 1_{Sin\_Accion\_Correc})$$

Con esta formula podemos asignarle un puntaje a cada establecimiento y a través de esto hacer rankings para priorizar.