# Modelos lineales generalizados

CEPAL

7/3/2022

# Lectura de la base

```
encuesta <- readRDS("../Data/encuesta.rds")
data("BigCity", package = "TeachingSampling")
```

# Definir diseño de la muestra con `srvyr`

```
library(srvyr)

diseno <- encuesta %>%
  as_survey_design(
    strata = Stratum,
    ids = PSU,
    weights = wk,
    nest = T
  )
```

# definir nuevas variables

```
diseno <- diseno %>% mutate(
  pobreza = ifelse(Poverty != "NotPoor", 1, 0),
  desempleo = ifelse(Employment == "Unemployed", 1, 0))
```

# Tablas de doble entrada para el tamaño

```
library(survey)
(tab_pobreza_sexo <- svyby(~factor(pobreza), ~Sex,
      FUN = svytotal, design = as.svrepdesign(diseno),
      se=F, na.rm=T, ci=T, keep.var=TRUE))
```

| | Sex | factor(pobreza)0 | factor(pobreza)1 | se1 | se2 |
|---|---|---|---|---|---|
| Female | Female | 47993 | 31197 | 2489 | 2352 |
| Male | Male | 38930 | 32146 | 2370 | 3588 |

```
(tab <- svytable(~pobreza + Sex, design = diseno))
```

| pobreza/Sex | Female | Male |
|---|---|---|
| 0 | 47993 | 38930 |
| 1 | 31197 | 32146 |

# Tablas de doble entrada para el proporción

```
(tab_pobreza_sexo <- svyby(~factor(pobreza), ~Sex,
      FUN = svymean, design = as.svrepdesign(diseno),
      se=F, na.rm=T, ci=T, keep.var=TRUE))
```

| | Sex | factor(pobreza)0 | factor(pobreza)1 | se1 | se2 |
|---|---|---|---|---|---|
| Female | Female | 0.6060 | 0.3940 | 0.0273 | 0.0273 |
| Male | Male | 0.5477 | 0.4523 | 0.0377 | 0.0377 |

```
prop.table(tab, margin = 2)
```

| pobreza/Sex | Female | Male |
|---|---|---|
| 0 | 0.606 | 0.5477 |
| 1 | 0.394 | 0.4523 |

# Prueba de independencia F

$$\hat{\pi}_{rc} = \frac{n_{r+}}{n_{++}} \times \frac{n_{+c}}{n_{++}}$$

$$\chi^2_{pearsom} = n_{++} \times \sum_r \sum_c \left( \frac{(p_{rc} - \hat{\pi}_{rc})^2}{\hat{\pi}_{rc}} \right)$$

$$G^2 = 2 \times n_{++} \times \sum_r \sum_c p_{cr} \times \ln \left( \frac{p_{rc}}{\hat{\pi}_{rc}} \right)$$

donde, $R$ es el número de filas y $C$ representa el número de columnas, la prueba tiene $(R-1) \times (C-1)$ grados de libertad.

# Prueba de independencia F

$$\chi^2_{(R-S)} = \chi^2_{(Pearson)} / GDEFF$$

$$G^2_{(R-S)} = G^2 / GDEFF$$

con $GDEFF$ el efecto generalizado del diseño, esta dado por

$$GDEFF = \frac{\sum_r \sum_c (1 - p_{rc})\, d^2(p_{rc}) - \sum_r (1 - p_{r+})\, d^2(p_{r+}) - \sum_c (1 - p_{+c})\, d^2(p_{+c})}{(R-1)(C-1)}$$

# Prueba de independencia F

$$F_{R-S,Pearson} = \chi^2_{R-S} / \left[ (R-1)(C-1) \right] \sim F_{(R-1)(C-1),(R-1)(C-1)df}$$

$$F_{R-S,LRT} = G^2_{R-S} / (C-1) \sim F_{(C-1),df}$$

donde $C$ es el número de columnas de la tabla cruzada

# Prueba de independencia ChiSq

```
summary(tab, statistic = "Chisq")
```

```
##          Sex
## pobreza Female  Male
##       0  47993 38930
##       1  31197 32146
##
##  Pearson's X^2: Rao & Scott adjustment
##
## data:  NextMethod()
## X-squared = 8.4, df = 1, p-value = 0.006
```

# Prueba de independencia F

```
summary(tab, statistic = "F")
```

```
##          Sex
## pobreza Female  Male
##       0  47993 38930
##       1  31197 32146
##
##  Pearson's X^2: Rao & Scott adjustment
##
## data:  NextMethod()
## F = 7.7, ndf = 1, ddf = 119, p-value = 0.007
```

# Estadístico de Wald

$$Q_{wald} = \hat{\boldsymbol{Y}}^t \left( \boldsymbol{H} \hat{\boldsymbol{V}} \left( \hat{\boldsymbol{N}} \right) \boldsymbol{H}^t \right)^{-1} \hat{\boldsymbol{Y}}$$

donde,

$$\hat{\boldsymbol{Y}} = \left( \hat{N} - E \right)$$

es un vector de $R \times C$ de diferencias entre los recuentos de celdas observadas y esperadas, esto es, $\hat{N}_{rc} - E_{rc}$

La matriz $\boldsymbol{H} \hat{\boldsymbol{V}} \left( \hat{\boldsymbol{N}} \right) \boldsymbol{H}^t$, representa la matriz de varianza-covarianza estimada para el vector de diferencias.

# Estadístico de Wald

La matriz $\boldsymbol{H}$ es la inversa de la matriz $\boldsymbol{J}$ dada por:

$$\boldsymbol{J} = -\left[\frac{\delta^2 \ln PL(\boldsymbol{B})}{\delta^2 \boldsymbol{B}}\right] \mid \boldsymbol{B} = \hat{\boldsymbol{B}}$$

$$\sum_h \sum_a \sum_i x_{hai}^t x_{hai} w_{hai} \hat{\pi}_{hai}(\boldsymbol{B})(1 - \hat{\pi}_{hai}(\boldsymbol{B}))$$

Bajo la hipótesis nula, el estadístico

$$Q_{wald} \sim \chi^2_{(R-1)\times(C-1)}$$

# Estadístico de Wald

$$F_{wald} = Q_{wald} \times \frac{df - (R-1)(C-1) + 1}{(R-1)(C-1)\, df} \sim F_{(R-1)(C-1),\, df-(R-1)(C-1)+1}$$

# Prueba de independencia Wald

```
summary(tab, statistic = "Wald")
```

```
##          Sex
## pobreza Female  Male
##       0  47993 38930
##       1  31197 32146
##
##  Design-based Wald test of association
##
## data:  NextMethod()
## F = 7.1, ndf = 1, ddf = 119, p-value = 0.009
```

# Prueba de independencia adjWald

```
summary(tab, statistic = "adjWald")
```

```
##          Sex
## pobreza Female  Male
##       0  47993 38930
##       1  31197 32146
##
##  Design-based Wald test of association
##
## data:  NextMethod()
## F = 7.1, ndf = 1, ddf = 119, p-value = 0.009
```

# Prueba de independencia lincom

```
summary(tab, statistic = "lincom")
```

```
##          Sex
## pobreza Female  Male
##       0  47993 38930
##       1  31197 32146
##
##  Pearson's X^2: asymptotic exact distribution
##
## data:  NextMethod()
## X-squared = 8.4, p-value = 0.007
```

# Prueba de independencia saddlepoint

```
summary(tab, statistic = "saddlepoint")
```

```
##          Sex
## pobreza Female  Male
##       0  47993 38930
##       1  31197 32146
##
##  Pearson's X^2: saddlepoint approximation
##
## data:  NextMethod()
## X-squared = 8.4, p-value = 0.007
```

# Modelo log lineal para tablas de contingencia

$$\log(p_{ijk}) = \mu + \lambda_i^X + \lambda_j^Y + \lambda_k^Z + \lambda_{ij}^{XY},$$

donde:

- $p_{ijk}$ = la proporción esperada en la celda bajo el modelo.
- $\mu = \log(p_0) = \frac{1}{\#\ de\ celdas}$

# Modelo log lineal para tablas de contingencia

```
mod1 <- svyloglin(~pobreza+Sex + pobreza:Sex , diseno)
(s1 <- summary(mod1))
```

```
## Loglinear model: svyloglin(~pobreza + Sex + pobreza:Sex
##                      coef      se        p
## pobreza1          0.15554 0.06368 0.014583
## Sex1              0.04483 0.02331 0.054403
## pobreza1:Sex1     0.05981 0.02163 0.005698
```

# Modelo log lineal para tablas de contingencia

```
mod2 <- svyloglin(~pobreza+Sex, diseno)
(s2 <- summary(mod2))
```

```
## Loglinear model: svyloglin(~pobreza + Sex, diseno)
##                 coef      se       p
## pobreza1 0.15822 0.06378 0.01311
## Sex1     0.05405 0.02454 0.02765
```

# Modelo log lineal para tablas de contingencia

```
anova(mod1, mod2)
```

```
## Analysis of Deviance Table
##  Model 1: y ~ pobreza + Sex
## Model 2: y ~ pobreza + Sex + pobreza:Sex
## Deviance= 8.422 p= 0.006573
## Score= 8.422 p= 0.006572
```

# Modelo de regresión logistica

$$g(\pi(x)) = logit(\pi(x)) = z = \ln\left(\frac{\pi(x)}{1-\pi(x)}\right) = B_0 + B_1 x_1 + \cdots + B_p x_p$$

$$\hat{\pi}(\boldsymbol{x}) = \frac{\exp\left(\boldsymbol{X}\hat{\boldsymbol{B}}\right)}{1-\exp\left(\boldsymbol{X}\hat{\boldsymbol{B}}\right)} = \frac{\exp\left(\hat{B}_0 + \hat{B}_1 x_1 + \cdots + \hat{B} x_p\right)}{1-\exp\left(\hat{B}_0 + \hat{B}_1 x_1 + \cdots + \hat{B} x_p\right)}$$

$$PL(\boldsymbol{B} \mid X) = \prod_{i=1}^{n} \left\{\pi(x_i)^{y_i} \left(1-\pi(x_i)\right)^{1-y_i}\right\}^{w_i}$$

con

$$\pi(x_i) = \frac{\exp(x_i \boldsymbol{B})}{1-\exp(x_i \boldsymbol{B})}$$

$$var\left(\hat{\boldsymbol{B}}\right) = \boldsymbol{J}^{-1} var\left(S\left(\hat{\boldsymbol{B}}\right)\right) \boldsymbol{J}^{-1}$$

# Modelo de regresión logistica

$$S(B) = \sum_{h} \sum_{a} \sum_{i} w_{hai} \boldsymbol{D}_{hai}^{t} \left[ \left( \pi_{hai} \left( \boldsymbol{B} \right) \right) \left( 1 - \pi_{hai} \left( \boldsymbol{B} \right) \right) \right]^{-1} \left( y_{hai} - \pi_{hai} \left( \boldsymbol{B} \right) \right) = 0$$

$$D_{hai} = \frac{\delta \left( \pi_{hai} \left( \boldsymbol{B} \right) \right)}{\delta B_j}$$

$j = 0, \ldots, p$

# Prueba de Wald para los parámetros del modelo

$$G = -2 \ln \left[ \frac{L\left(\hat{\beta}_{MLE}\right)_{reduced}}{L\left(\hat{\beta}_{MLE}\right)_{full}} \right]$$

$$\hat{\psi} = \exp\left(\hat{B}_1\right)$$

$$CI\left(\psi\right) = \exp\left(\hat{B}_j \pm t_{df,1-\frac{\alpha}{2}} se\left(\hat{B}_j\right)\right)$$

# Tablas de contingencia

| Sex | pobreza | se | ci_l | ci_u |
|---|---|---|---|---|
| Female | 0.3940 | 0.0273 | 0.3405 | 0.4474 |
| Male | 0.4523 | 0.0377 | 0.3784 | 0.5262 |

| Zone | pobreza | se | ci_l | ci_u |
|---|---|---|---|---|
| Rural | 0.4498 | 0.0551 | 0.3417 | 0.5579 |
| Urban | 0.3955 | 0.0307 | 0.3352 | 0.4557 |

| Region | pobreza | se | ci_l | ci_u |
|---|---|---|---|---|
| Norte | 0.5882 | 0.0571 | 0.4762 | 0.7002 |
| Sur | 0.3917 | 0.0594 | 0.2752 | 0.5082 |
| Centro | 0.2370 | 0.0437 | 0.1514 | 0.3227 |
| Occidente | 0.3602 | 0.0435 | 0.2750 | 0.4455 |
| Oriente | 0.4978 | 0.0929 | 0.3156 | 0.6799 |

# Prueba de independencia ChiSq

```
## Pearson's X^2: Rao & Scott adjustment
pobreza_sex <-  svychisq(
  formula = ~pobreza + Sex, design = diseno)
tidy( pobreza_sex)
```

| ndf | ddf | statistic | p.value | method |
|---|---|---|---|---|
| 1 | 119 | 7.653 | 0.0066 | Pearson's X^2: Rao & Scott adjustment |

# Prueba de independencia ChiSq

```
pobreza_Zona <- svychisq(
  formula = ~pobreza + Zone, design = diseno)
tidy(pobreza_Zona) %>% select(-method)
```

| ndf | ddf | statistic | p.value |
|---|---|---|---|
| 1 | 119 | 0.7523 | 0.3875 |

# Prueba de independencia ChiSq

```
pobreza_Region <- svychisq(
  formula = ~pobreza + Region, design = diseno)
   tidy(pobreza_Region) %>% select(-method)
```

| ndf | ddf | statistic | p.value |
|---|---|---|---|
| 2.885 | 343.3 | 4.221 | 0.0067 |

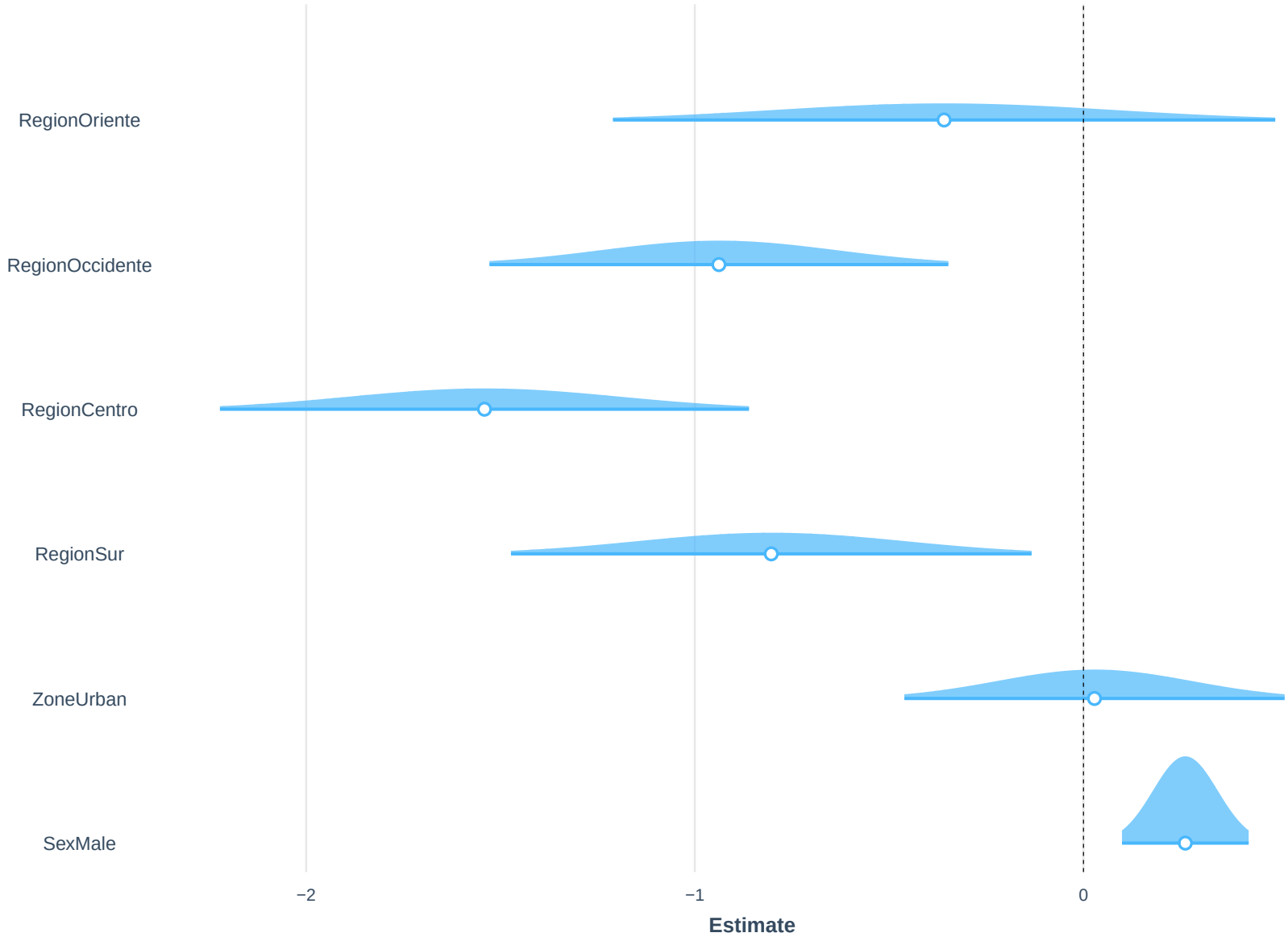
# Modelo log lineal ajustado

```
mod_loglin <- svyglm(
  pobreza ~ Sex + Zone + Region,
   family=quasibinomial, design=diseno)
tidy(mod_loglin)
```

| term | estimate | std.error | statistic | p.value |
|---|---|---|---|---|
| (Intercept) | 0.2202 | 0.2519 | 0.8743 | 0.3838 |
| SexMale | 0.2621 | 0.0832 | 3.1515 | 0.0021 |
| ZoneUrban | 0.0283 | 0.2496 | 0.1134 | 0.9099 |
| RegionSur | -0.8033 | 0.3418 | -2.3505 | 0.0205 |
| RegionCentro | -1.5413 | 0.3473 | -4.4378 | 0.0000 |
| RegionOccidente | -0.9381 | 0.3013 | -3.1140 | 0.0023 |
| RegionOriente | -0.3586 | 0.4348 | -0.8249 | 0.4112 |

# Plot de la distribución de los betas

```
plot_summs(mod_loglin,
           scale = TRUE, plot.distributions = TRUE)
```



# Modelo log lineal ajustado

```
bind_cols(
data.frame(exp_estimado = exp(coef(mod_loglin))),
as.data.frame(exp(confint(mod_loglin)))
)
```

| | exp_estimado | 2.5 % | 97.5 % |
|---|---|---|---|
| (Intercept) | 1.2464 | 0.7567 | 2.0530 |
| SexMale | 1.2997 | 1.1022 | 1.5325 |
| ZoneUrban | 1.0287 | 0.6274 | 1.6868 |
| RegionSur | 0.4478 | 0.2275 | 0.8814 |
| RegionCentro | 0.2141 | 0.1076 | 0.4260 |
| RegionOccidente | 0.3914 | 0.2155 | 0.7109 |
| RegionOriente | 0.6986 | 0.2952 | 1.6532 |

# Estadístico de Wald sobre los parámetros

```
regTermTest(model = mod_loglin, ~Sex)
```

```
## Wald test for Sex
##  in svyglm(formula = pobreza ~ Sex + Zone + Region, desi
##      family = quasibinomial)
## F =  9.932  on  1  and  113  df: p= 0.0021
```

```
regTermTest(model = mod_loglin, ~Zone)
```

```
## Wald test for Zone
##  in svyglm(formula = pobreza ~ Sex + Zone + Region, desi
##      family = quasibinomial)
## F =  0.01286  on  1  and  113  df: p= 0.91
```

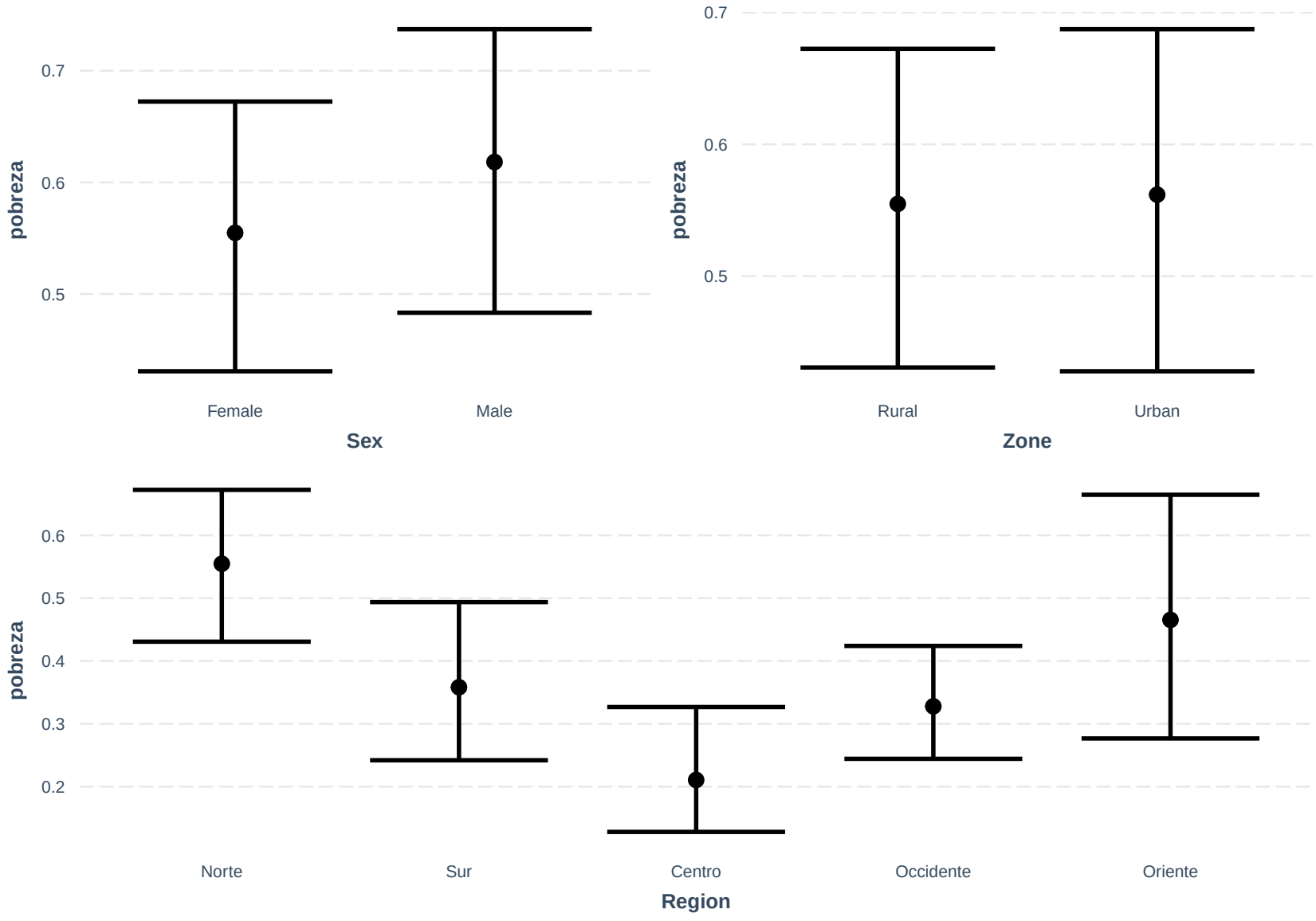
```
regTermTest(model = mod_loglin, ~Region)
```

```
## Wald test for Region
##  in svyglm(formula = pobreza ~ Sex + Zone + Region, desi
##      family = quasibinomial)
## F =  5.957  on  4  and  113  df: p= 0.00022
```

# Efecto del modelo.

```
effe_sex <- effect_plot(mod_loglin, pred = Sex,
                        interval = TRUE)
effe_Zona <-effect_plot(mod_loglin, pred = Zone,
                        interval = TRUE)
effe_Region <- effect_plot(mod_loglin, pred = Region,
                           interval = TRUE)
(effe_sex |effe_Zona)/effe_Region
```

# Efecto del modelo.



# Modelo log lineal ajustado con interacciones

```
mod_loglin_int <- svyglm(pobreza ~ Sex + Zone + Region +
                                   Sex:Zone + Sex:Region,
                         family=quasibinomial, design=diseno)
tab_mod <- tidy(mod_loglin_int) %>% arrange(p.value)
tab_mod %>% slice(1:6)
```

| term | estimate | std.error | statistic | p.value |
|---|---|---|---|---|
| RegionCentro | -1.7878 | 0.3336 | -5.359 | 0.0000 |
| RegionOccidente | -1.0489 | 0.2696 | -3.891 | 0.0002 |
| RegionSur | -0.8349 | 0.3301 | -2.529 | 0.0129 |
| SexMale:RegionCentro | 0.4813 | 0.3164 | 1.521 | 0.1311 |
| (Intercept) | 0.3128 | 0.2196 | 1.424 | 0.1573 |
| SexMale:RegionOriente | 0.3323 | 0.2456 | 1.353 | 0.1789 |

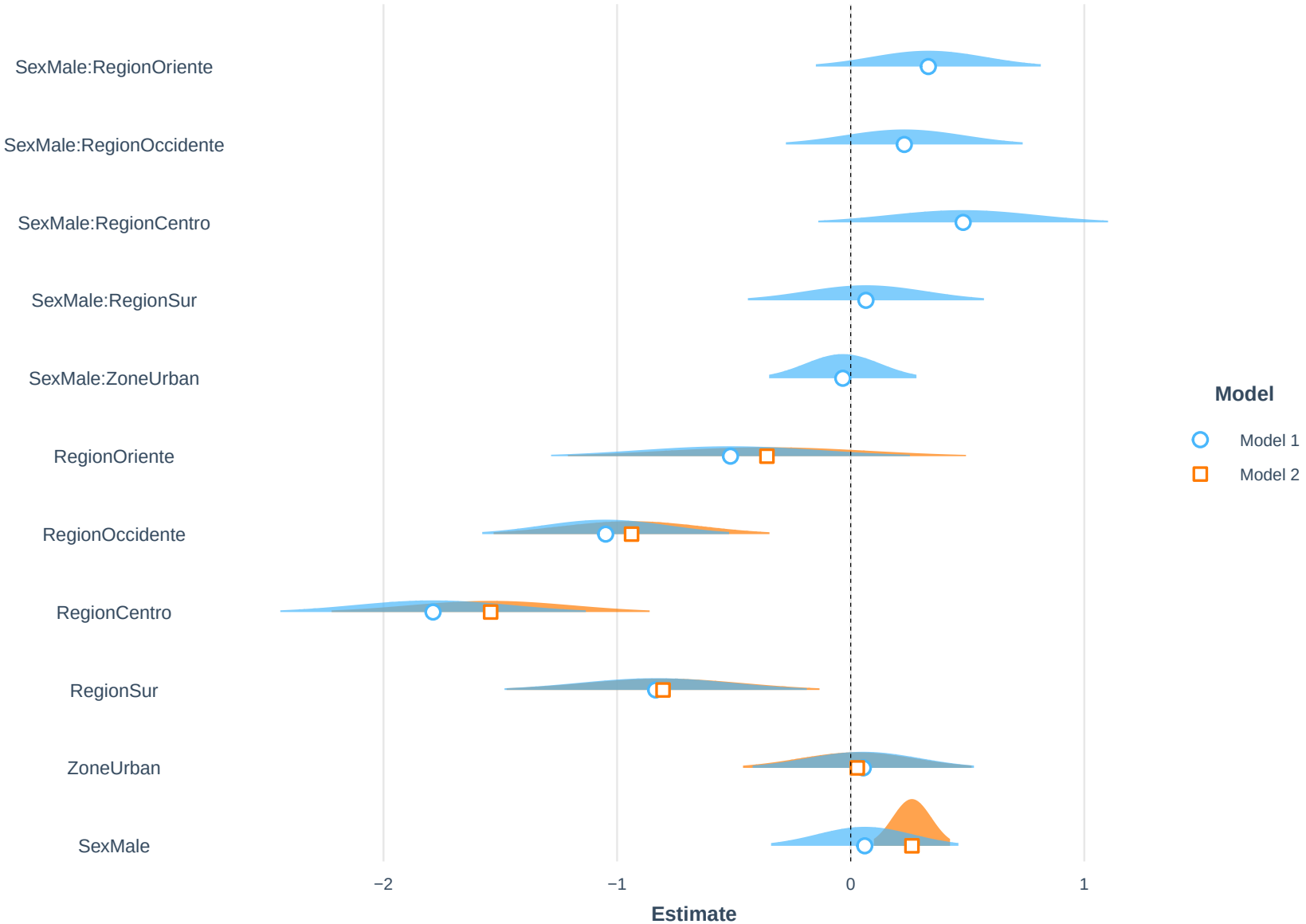
# Modelo log lineal ajustado con interacciones

```
tab_mod %>% slice(7:12)
```

| term | estimate | std.error | statistic | p.value |
|---|---|---|---|---|
| RegionOriente | -0.5146 | 0.3918 | -1.3134 | 0.1918 |
| SexMale:RegionOccidente | 0.2294 | 0.2586 | 0.8872 | 0.3770 |
| SexMale | 0.0600 | 0.2046 | 0.2935 | 0.7697 |
| SexMale:RegionSur | 0.0652 | 0.2578 | 0.2527 | 0.8010 |
| ZoneUrban | 0.0539 | 0.2417 | 0.2232 | 0.8238 |
| SexMale:ZoneUrban | -0.0339 | 0.1609 | -0.2104 | 0.8337 |

# Plot de la distribución de los betas

```
plot_summs(mod_loglin_int, mod_loglin, scale = TRUE, plot.c
```



# Modelo log lineal ajustado

| | exp_estimado | 2.5 % | 97.5 % |
|---|---|---|---|
| (Intercept) | 1.3672 | 0.8846 | 2.1131 |
| SexMale | 1.0619 | 0.7079 | 1.5929 |
| ZoneUrban | 1.0554 | 0.6537 | 1.7039 |
| RegionSur | 0.4339 | 0.2255 | 0.8348 |
| RegionCentro | 0.1673 | 0.0864 | 0.3242 |
| RegionOccidente | 0.3503 | 0.2053 | 0.5978 |
| RegionOriente | 0.5977 | 0.2749 | 1.2996 |
| SexMale:ZoneUrban | 0.9667 | 0.7027 | 1.3298 |
| SexMale:RegionSur | 1.0673 | 0.6403 | 1.7792 |
| SexMale:RegionCentro | 1.6182 | 0.8643 | 3.0300 |
| SexMale:RegionOccidente | 1.2578 | 0.7534 | 2.1000 |
| SexMale:RegionOriente | 1.3942 | 0.8568 | 2.2687 |

# Estadístico de Wald sobre los parámetros

```
regTermTest(model = mod_loglin_int, ~Sex)
```

```
## Wald test for Sex
##  in svyglm(formula = pobreza ~ Sex + Zone + Region + Sex
##      design = diseno, family = quasibinomial)
## F =  0.08615  on  1  and  108  df: p= 0.77
```

```
regTermTest(model = mod_loglin_int, ~Zone)
```

```
## Wald test for Zone
##  in svyglm(formula = pobreza ~ Sex + Zone + Region + Sex
##      design = diseno, family = quasibinomial)
## F =  0.0498  on  1  and  108  df: p= 0.82
```

```
regTermTest(model = mod_loglin_int, ~Region)
```

```
## Wald test for Region
##  in svyglm(formula = pobreza ~ Sex + Zone + Region + Sex
##      design = diseno, family = quasibinomial)
## F =  8.719  on  4  and  108  df: p= 3.9e-06
```

# Estadístico de Wald sobre los parámetros

```
regTermTest(model = mod_loglin_int, ~Sex:Zone)
```

```
## Wald test for Sex:Zone
##  in svyglm(formula = pobreza ~ Sex + Zone + Region + Sex
##     design = diseno, family = quasibinomial)
## F =  0.04428  on  1  and  108  df: p= 0.83
```

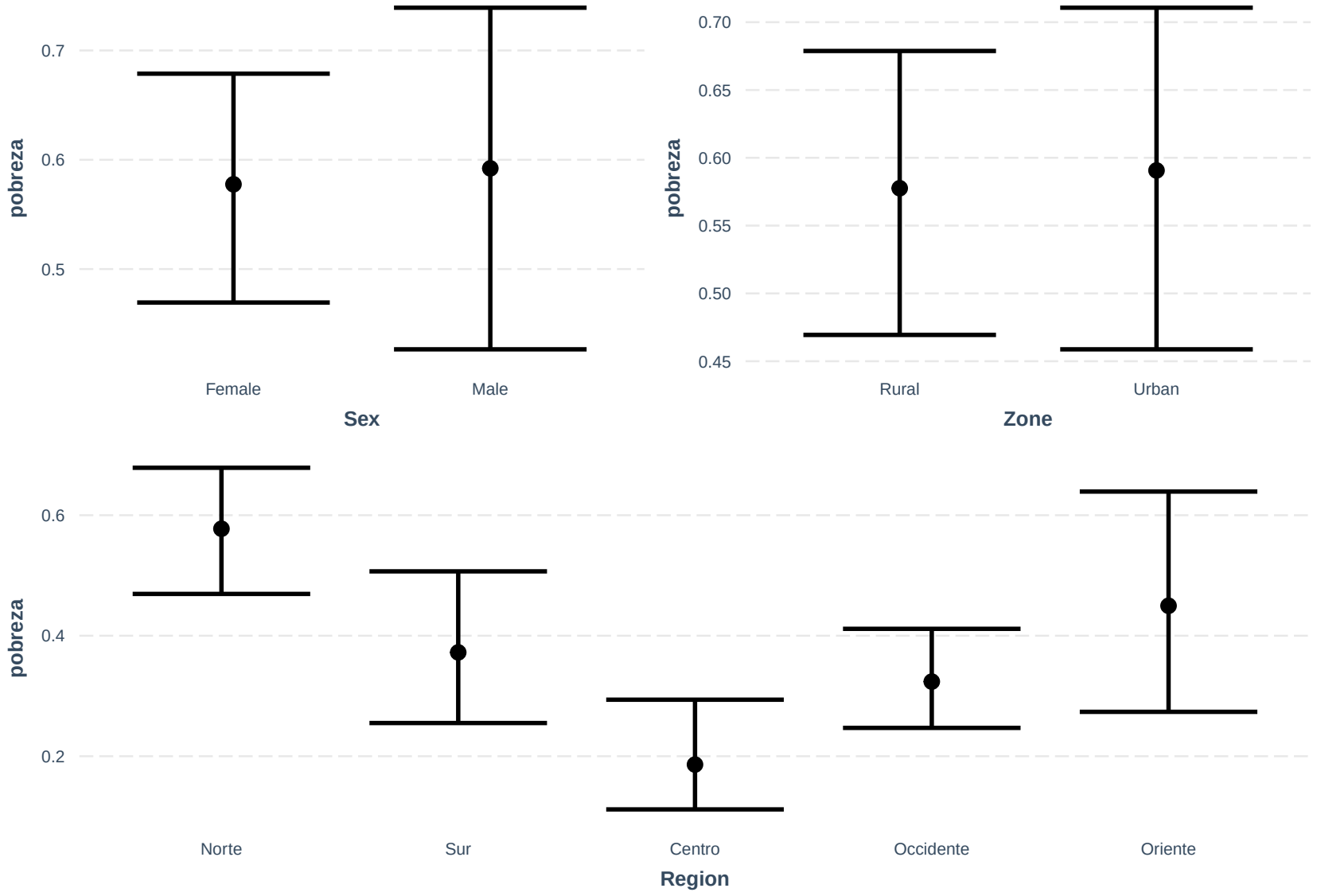
```
regTermTest(model = mod_loglin_int, ~Sex:Region)
```

```
## Wald test for Sex:Region
##  in svyglm(formula = pobreza ~ Sex + Zone + Region + Sex
##     design = diseno, family = quasibinomial)
## F =  0.8904  on  4  and  108  df: p= 0.47
```

# Efecto del modelo.

```
effe_sex <- effect_plot(mod_loglin_int, pred = Sex,
                        interval = TRUE)
effe_Zona <-effect_plot(mod_loglin_int, pred = Zone,
                        interval = TRUE)
effe_Region <- effect_plot(mod_loglin_int, pred = Region,
                           interval = TRUE)
(effe_sex |effe_Zona)/effe_Region
```

# Efecto del modelo.



# Modelo log lineal ajustado con Q_Weighting

```
fit_wgt <- lm(wk ~  Sex + Zone + Region , data = encuesta)
wgt_hat <- predict(fit_wgt)
encuesta %<>% mutate(wk2 = wk/wgt_hat)

diseno_qwgt <- encuesta %>%
  as_survey_design(
    strata = Stratum,
    ids = PSU,
    weights = wk2,
    nest = T
  )
```

# Modelo log lineal ajustado con Q_Weighting

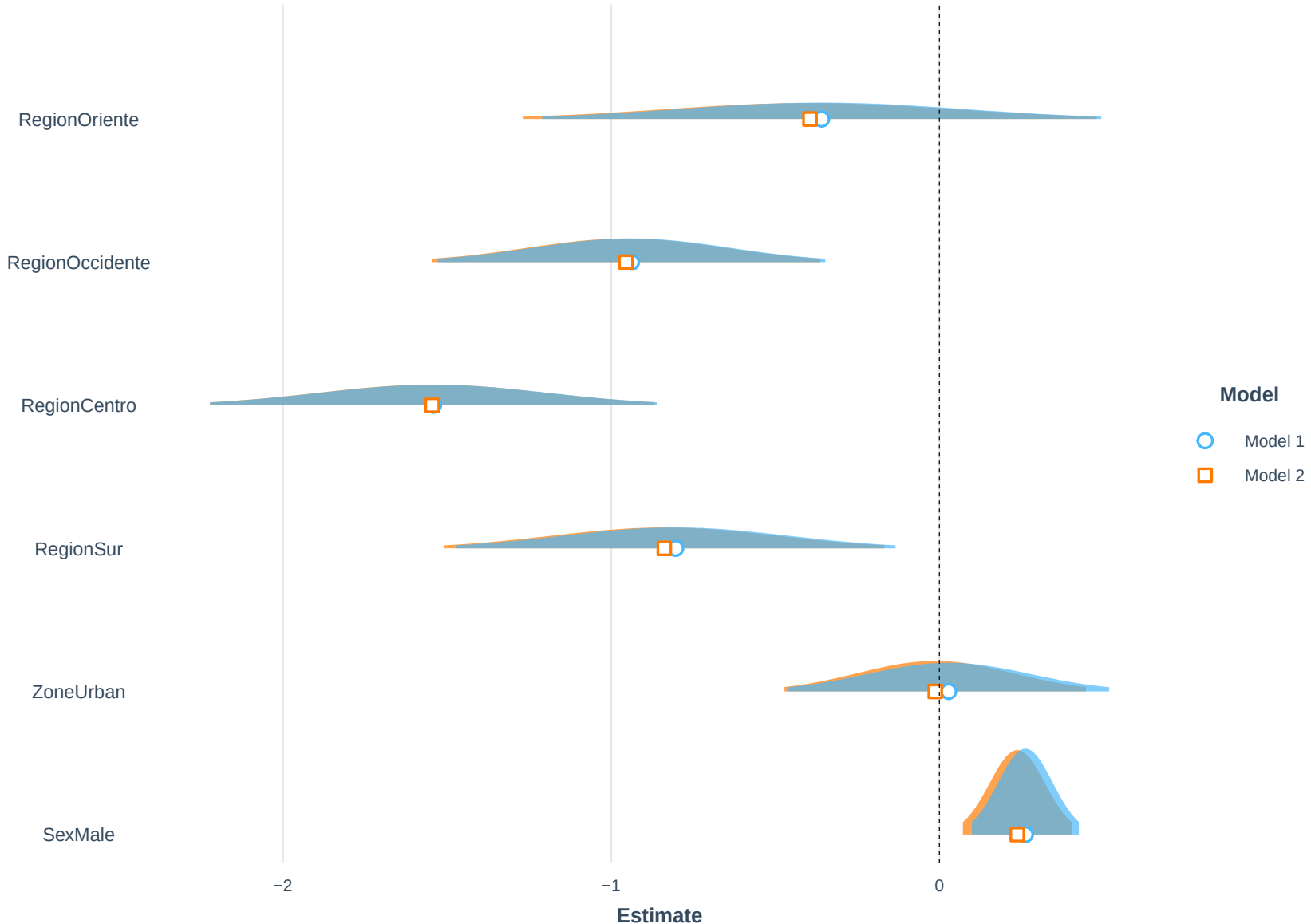
```
diseno_qwgt <- diseno_qwgt %>% mutate(
  pobreza = ifelse(Poverty != "NotPoor", 1, 0))

mod_loglin_qwgt <- svyglm(pobreza ~ Sex + Zone + Region,
                          family=quasibinomial,
                          design=diseno_qwgt)
(tab_mod <- tidy(mod_loglin_qwgt) )
```

# Modelo log lineal ajustado con Q_Weighting

| term | estimate | std.error | statistic | p.value |
|---|---|---|---|---|
| (Intercept) | 0.2723 | 0.2492 | 1.0928 | 0.2768 |
| SexMale | 0.2374 | 0.0846 | 2.8068 | 0.0059 |
| ZoneUrban | -0.0125 | 0.2343 | -0.0535 | 0.9574 |
| RegionSur | -0.8380 | 0.3423 | -2.4485 | 0.0159 |
| RegionCentro | -1.5450 | 0.3455 | -4.4720 | 0.0000 |
| RegionOccidente | -0.9547 | 0.3020 | -3.1617 | 0.0020 |
| RegionOriente | -0.3941 | 0.4457 | -0.8843 | 0.3784 |

# Plot de la distribución de los betas

```
plot_summs(mod_loglin, mod_loglin_qwgt,
           scale = TRUE, plot.distributions = TRUE)
```



# Modelo log lineal ajustado

| | exp_estimado | 2.5 % | 97.5 % |
|---|---|---|---|
| (Intercept) | 1.3130 | 0.8014 | 2.1511 |
| SexMale | 1.2679 | 1.0723 | 1.4993 |
| ZoneUrban | 0.9875 | 0.6208 | 1.5710 |
| RegionSur | 0.4326 | 0.2196 | 0.8522 |
| RegionCentro | 0.2133 | 0.1076 | 0.4229 |
| RegionOccidente | 0.3849 | 0.2116 | 0.7002 |
| RegionOriente | 0.6743 | 0.2788 | 1.6305 |

# Estadístico de Wald sobre los parámetros

```
regTermTest(model = mod_loglin_qwgt, ~Sex)
```

```
## Wald test for Sex
##  in svyglm(formula = pobreza ~ Sex + Zone + Region, desi
##      family = quasibinomial)
## F =  7.878  on  1  and  113  df: p= 0.0059
```

```
regTermTest(model = mod_loglin_qwgt, ~Zone)
```

```
## Wald test for Zone
##  in svyglm(formula = pobreza ~ Sex + Zone + Region, desi
##      family = quasibinomial)
## F =  0.002866  on  1  and  113  df: p= 0.96
```

```
regTermTest(model = mod_loglin_qwgt, ~Region)
```

```
## Wald test for Region
##  in svyglm(formula = pobreza ~ Sex + Zone + Region, desi
##      family = quasibinomial)
## F =  5.87  on  4  and  113  df: p= 0.00025
```

# Efecto del modelo.

```
effe_sex <- effect_plot(mod_loglin_qwgt, pred = Sex,
                        interval = TRUE)
effe_Zona <-effect_plot(mod_loglin_qwgt, pred = Zone,
                        interval = TRUE)
effe_Region <- effect_plot(mod_loglin_qwgt, pred = Region,
                           interval = TRUE)
(effe_sex |effe_Zona)/effe_Region
```

# Efecto del modelo.