

BIOS 7345: Advanced Regression Analysis I

Andrew J. Spieker, Ph.D.

Assistant Professor of Biostatistics
Vanderbilt University Medical Center

Set 16: Hypothesis testing for GLMs

Version: 11/10/2023

TABLE OF CONTENTS

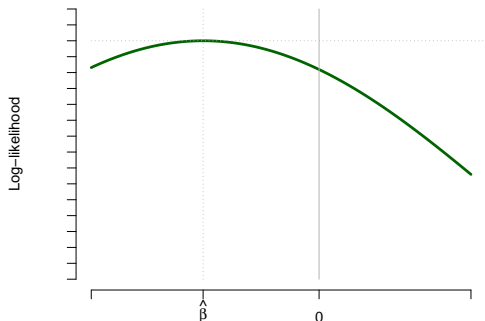
- 1 Hypothesis testing
- 2 Examples: Logistic regression
- 3 Examples: Gaussian regression
- 4 Inverting score tests

Generally:

- We seek to test hypotheses $H_0 : \mathbf{C}\boldsymbol{\beta} = \mathbf{0}$, where $\boldsymbol{\beta}$ denotes the regression parameters of a GLM (you may be thinking about it through the likelihood, quasi-likelihood, or estimating equations framework) and $\mathbf{C} = \mathbf{C}_{Q \times K}$ encodes Q linear hypotheses.
- The goal in this set of notes is to present and discuss different hypothesis testing methods. You'll already be familiar with many of the ideas. The three major classes of tests we will discuss are:
 - ▶ Likelihood ratio tests.
 - ▶ Score-based tests (also called Rao tests).
 - ▶ Wald-based tests.
- In correctly specified models, and under suitable regularity conditions, the three are asymptotically equivalent. This can be seen graphically.

HYPOTHESIS TESTING

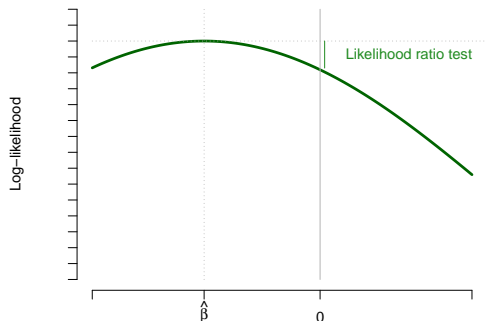
In pictures: Log-likelihood



- In our settings, the log-likelihood should generally be concave in a neighborhood of $\hat{\beta}$.

HYPOTHESIS TESTING

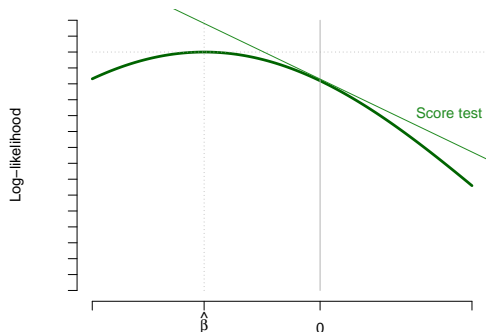
In pictures: Likelihood-ratio based tests



- A likelihood ratio test compares the maximum log-likelihood to that under the null.

HYPOTHESIS TESTING

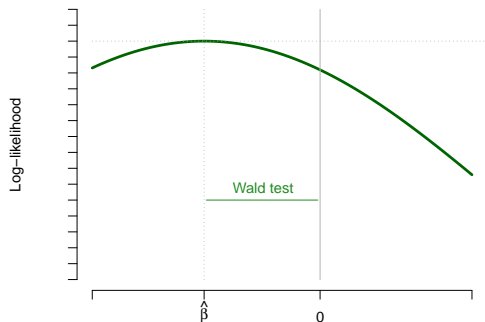
In pictures: Score-based tests



- A score-based test involves evaluating the slope of the log-likelihood under the null.

HYPOTHESIS TESTING

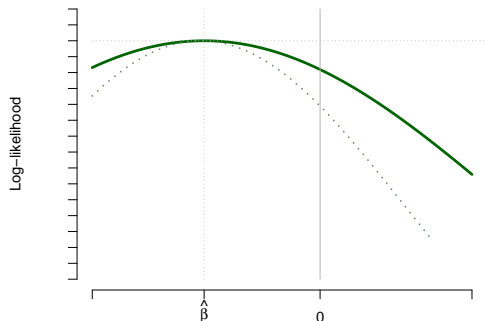
In pictures: Wald-based tests



- A Wald-based test involves evaluating how far the observed value $\hat{\beta}$ is from the null (relative to variability).

HYPOTHESIS TESTING

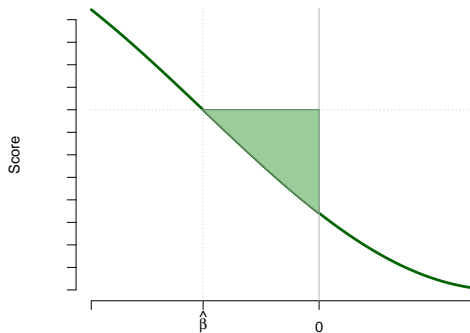
In pictures: Increasing sample size



- Greater concavity in the log-likelihood. What are the consequences for each of the tests? What are the implications if H_0 is true?

HYPOTHESIS TESTING

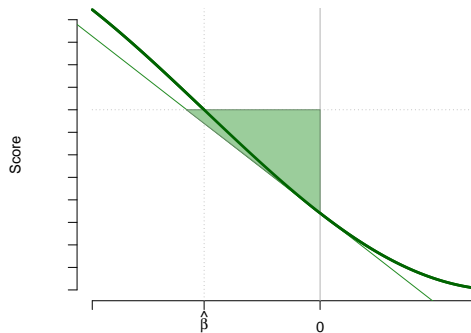
In pictures: Likelihood ratio-based tests



- The likelihood ratio test statistic is twice the shaded area.

HYPOTHESIS TESTING

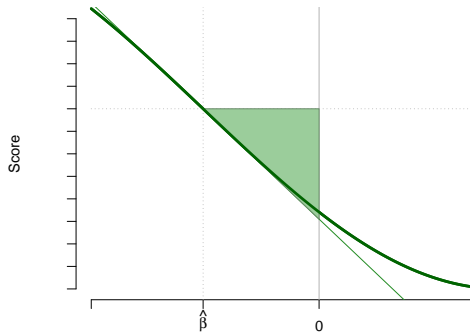
In pictures: Score-based tests



- The score statistic is twice the shaded area.

HYPOTHESIS TESTING

In pictures: Wald-based tests



- The Wald statistic is twice the shaded area.

For GLMs: Wald statistic

- The following is a form for the Wald statistic (we've encountered it before):

$$\chi_W^2 = (\mathbf{C}\hat{\boldsymbol{\beta}})^T \left(\mathbf{C}\widehat{\text{Cov}}[\hat{\boldsymbol{\beta}}]\mathbf{C}^T \right)^{-1} (\mathbf{C}\hat{\boldsymbol{\beta}}).$$

- $\widehat{\text{Cov}}[\hat{\boldsymbol{\beta}}]$ is any reasonable and appropriate estimator (model-based, sandwich, quasi-likelihood, bootstrap-based).
- Under H_0 , we have $\chi_W^2 \rightarrow_d \chi_Q^2$ (I state this without proof; we have already done a rigorous treatment of hypothesis testing for linear regression, so I don't wish to belabor the point).

For GLMs: Score statistic

- Let $\hat{\boldsymbol{\beta}}^0$ denote the estimate under H_0 .
 - ▶ Subject to the constraint $\mathbf{C}\boldsymbol{\beta} = \mathbf{0}$.
- The score takes the form $\mathbb{S}_N(\boldsymbol{\beta}, \boldsymbol{\phi}) = \mathbf{D}^T \mathbf{V}^{-1}(\mathbf{y} - \boldsymbol{\mu})/\boldsymbol{\phi}$.
- The information takes the form $\mathcal{I}_N(\boldsymbol{\beta}, \boldsymbol{\phi}) = \mathbb{A}_N(\boldsymbol{\beta})/\boldsymbol{\phi}$.
- The following is a form for the score statistic:

$$X_S^2 = (\mathbb{S}_N(\hat{\boldsymbol{\beta}}^0, \hat{\boldsymbol{\phi}}))^T \left(\mathcal{I}_N(\hat{\boldsymbol{\beta}}^0, \hat{\boldsymbol{\phi}}) \right)^{-1} (\mathbb{S}_N(\hat{\boldsymbol{\beta}}^0, \hat{\boldsymbol{\phi}})).$$

- Importantly, $\hat{\boldsymbol{\phi}}$ denotes the estimate **under the full model**.
- This test is model-based, although use of the *observed* information may offer some robustness under certain circumstances.
 - ▶ This is a rabbit hole I'm electing not to go down.
- Under H_0 , we have $X_S^2 \rightarrow_d \chi_Q^2$.

For GLMs: Likelihood ratio test statistic

- Let $\hat{\boldsymbol{\beta}}^0$ denote the estimate under H_0 .
 - ▶ Subject to the constraint $\mathbf{C}\boldsymbol{\beta} = \mathbf{0}$.
- The following is a form for the likelihood ratio test statistic:

$$X_{LR}^2 = -2 \left(\ell(\hat{\boldsymbol{\beta}}^0, \hat{\boldsymbol{\phi}}) - \ell(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\phi}}) \right)$$

- Importantly, $\hat{\boldsymbol{\phi}}$ denotes the estimate **under the full model**.
- This test is model-based.
- Under H_0 , we have $X_{LR}^2 \rightarrow_d \chi_Q^2$.

TABLE OF CONTENTS

- 1 Hypothesis testing
- 2 Examples: Logistic regression
- 3 Examples: Gaussian regression
- 4 Inverting score tests

Model setup: For simulation

- Exposure: $X = 0, 1, 2$, each with probability $1/3$.
- Outcome: $Y \sim \text{Bernoulli}(p = \text{expit}(\beta_0 + \beta_1 1(x = 1) + \beta_2 1(x = 2)))$.
- Let $\beta_0 = -1$, $\beta_1 = 0.4$ and $\beta_2 = 0.6$.
- Let $N = 150$.
- Consider the following hypothesis tests:
 - 1 $H_0 : \beta_1 = 0$ (groups 0 and 1 the same).
 - 2 $H_0 : \beta_2 - \beta_1 = 0$ (groups 1 and 2 the same).
 - 3 $H_0 : \beta_1 = \beta_2 = 0$ (all three groups the same).

Simulated example: Generate data

```
## Important function
expit <- function(x)
{
  expitx <- exp(x)/(1 + exp(x))
  return(expitx)
}

## Set seed and generate data
set.seed(7345)
n <- 150
beta <- matrix(c(-1, 0.4, 0.6), nrow = 3)
x <- sample(c(0,1,2), size = n, replace = TRUE)
X <- matrix(cbind(1, as.numeric(x==1), as.numeric(x==2)), ncol = 3)
y <- rbinom(n, 1, expit(X %*% beta))
```

EXAMPLES: LOGISTIC REGRESSION

Simulated example: Fit GLM (unrestricted)

```
## Initialize and run GLM
betaj <- c(0,0,0)
tol <- 1
iter <- 1

while(tol > 1e-15 & iter < 50)
{
  betaj.prior <- betaj
  etaj <- c(X %*% betaj)
  Gn <- t(X) %*% (y - expit(etaj))
  An <- t(X) %*% diag(expit(etaj)*(1 - expit(etaj))) %*% X
  betaj <- betaj + solve(An) %*% Gn
  tol <- sum((betaj - betaj.prior)^2)
  iter <- iter + 1
}
bhat <- betaj
Vhat <- solve(An)

> c(bhat)
[1] -1.0726368  0.2841794  0.7159619
```

Test 1: $H_0 : \beta_1 = 0$ (groups 0 and 1 the same).

- Model: $Y \sim \text{Bernoulli}(p = \text{expit}(\beta_0 + \beta_1 1(x = 1) + \beta_2 1(x = 2)))$.
- Constrained model: $Y \sim \text{Bernoulli}(p = \text{expit}(\beta_0 + \beta_2 1(x = 2)))$.

EXAMPLES: LOGISTIC REGRESSION

Simulated example: Fit GLM (constrained)

```
X0 <- cbind(X[,1], X[,3])
betaj <- c(0,0)
tol <- 1
iter <- 1

while(tol > 1e-15 & iter < 50)
{
  betaj.prior <- betaj
  etaj <- c(X0 %*% betaj)
  Gn <- t(X0) %*% (y - expit(etaj))
  An <- t(X0) %*% diag(expit(etaj)*(1 - expit(etaj))) %*% X0
  betaj <- betaj + solve(An) %*% Gn
  tol <- sum((betaj - betaj.prior)^2)
  iter <- iter + 1
}

bhat.0 <- cbind(c(betaj[1], 0, betaj[2]))

> c(bhat.0)
[1] -0.9304754  0.0000000  0.5738004
```

Simulated example: Likelihood ratio test

```
loglik <- sum(log(dbinom(y, size = 1, prob = expit(X %*% bhat))))  
loglik.0 <- sum(log(dbinom(y, size = 1, prob = expit(X %*% bhat.0))))  
LR <- -2*(loglik.0 - loglik)  
pLR <- 1 - pchisq(LR, df = 1)
```

Simulated example: Score test

```
Gn0 <- t(X) %*% (y - expit(X %*% bhat.0))
An0 <- t(X) %*% diag(c(expit(X %*% bhat.0)*
                      (1 - expit(X %*% bhat.0)))) %*% X
S <- t(Gn0) %*% solve(An0) %*% Gn0
pS <- 1 - pchisq(S, df = 1)
```

Simulated example: Wald test

```
C <- matrix(0, nrow = 1, ncol = 3)
C[1,2] <- 1
```

```
> C
      [,1] [,2] [,3]
[1,]    0    1    0
```

```
W <- t(C %*% bhat) %*% solve(C %*% Vhat %*% t(C)) %*% (C %*% bhat)
pW <- 1 - pchisq(W, df = 1)
```

Simulated example: Report and compare to ANOVA function

```
> pLR
[1] 0.5247948

> pS
      [,1]
[1,] 0.5248092

> pW
      [,1]
[1,] 0.5253544

zz.Full <- glm(y ~ X[,2] + X[,3], family = binomial, control = list(epsilon = 1e-15))
zz.Reduced <- glm(y ~ X[,3], family = binomial, control = list(epsilon = 1e-15))

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "LRT")$Pr[2]
[1] 0.5247948

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "Rao")$Pr[2]
[1] 0.5248092

## Not aware of a Wald-based test from anova() function
```


Test 2: $H_0 : \beta_2 - \beta_1 = 0$ (groups 1 and 2 the same).

- Model: $Y \sim \text{Bernoulli}(p = \text{expit}(\beta_0 + \beta_1 1(x = 1) + \beta_2 1(x = 2)))$.
- Constrained model: $Y \sim \text{Bernoulli}(p = \text{expit}(\beta_0 + \beta_1 1(x > 0)))$.

EXAMPLES: LOGISTIC REGRESSION

Simulated example: Fit GLM (constrained)

```
X0 <- cbind(X[,1], X[,2] + X[,3])
betaj <- c(0,0)
tol <- 1
iter <- 1

while(tol > 1e-15 & iter < 50)
{
  betaj.prior <- betaj
  etaj <- c(X0 %*% betaj)
  Gn <- t(X0) %*% (y - expit(etaj))
  An <- t(X0) %*% diag(expit(etaj)*(1 - expit(etaj))) %*% X0
  betaj <- betaj + solve(An) %*% Gn
  tol <- sum((betaj - betaj.prior)^2)
  iter <- iter + 1
}

bhat.0 <- cbind(c(betaj[1], betaj[2], betaj[2]))

> c(bhat.0)
[1] -1.072637  0.513021  0.513021
```

Simulated example: Wald test

```
C <- matrix(0, nrow = 1, ncol = 3)
C[1,2] <- -1
C[1,3] <- 1
```

```
> C
      [,1] [,2] [,3]
[1,]    0  -1    1
```

```
W <- t(C %*% bhat) %*% solve(C %*% Vhat %*% t(C)) %*% (C %*% bhat)
pW <- 1 - pchisq(W, df = 1)
```

Simulated example: Report and compare to ANOVA function

```
## The likelihood ratio test has the same code as previously shown
> pLR
[1] 0.3039759

## The score test has the same code as previously shown
> pS
      [,1]
[1,] 0.3048367

> pW
      [,1]
[1,] 0.3060022

zz.Full <- glm(y ~ X[,2] + X[,3], family = binomial, control = list(epsilon = 1e-15))
zz.Reduced <- glm(y ~ I(X[,2] + X[,3]), family = binomial, control = list(epsilon = 1e-15))

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "LRT")$Pr[2]
[1] 0.3039759

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "Rao")$Pr[2]
[1] 0.3048367
```

Test 3: $H_0 : \beta_1 = \beta_2 = 0$ (all groups the same).

- Model: $Y \sim \text{Bernoulli}(p = \text{expit}(\beta_0 + \beta_1 1(x = 1) + \beta_2 1(x = 2)))$.
- Constrained model: $Y \sim \text{Bernoulli}(p = \text{expit}(\beta_0))$.

EXAMPLES: LOGISTIC REGRESSION

Simulated example: Fit GLM (constrained)

```
X0 <- cbind(X[,1])
betaj <- c(0)
tol <- 1
iter <- 1

while(tol > 1e-15 & iter < 50)
{
  betaj.prior <- betaj
  etaj <- c(X0 %*% betaj)
  Gn <- t(X0) %*% (y - expit(etaj))
  An <- t(X0) %*% diag(expit(etaj)*(1 - expit(etaj))) %*% X0
  betaj <- betaj + solve(An) %*% Gn
  tol <- sum((betaj - betaj.prior)^2)
  iter <- iter + 1
}

bhat.0 <- cbind(c(betaj[1], 0, 0))

> c(bhat.0)
[1] -0.7233002  0.0000000  0.0000000
```

Simulated example: Wald test

```
C <- matrix(0, nrow = 2, ncol = 3)
C[1,2] <- 1
C[2,3] <- 1
```

```
> C
      [,1] [,2] [,3]
[1,]    0    1    0
[2,]    0    0    1
```

```
W <- t(C %*% bhat) %*% solve(C %*% Vhat %*% t(C)) %*% (C %*% bhat)
pW <- 1 - pchisq(W, df = 2)
```

Simulated example: Report and compare to ANOVA function

```
## The likelihood ratio test has the same code as previously shown (except df=2)
> pLR
[1] 0.2336163

## The score test has the same code as previously shown (except df=2)
> pS
      [,1]
[1,] 0.232581

> pW
      [,1]
[1,] 0.2370055

zz.Full <- glm(y ~ X[,2] + X[,3], family = binomial, control = list(epsilon = 1e-15))
zz.Reduced <- glm(y ~ 1, family = binomial, control = list(epsilon = 1e-15))

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "LRT")$Pr[2]
[1] 0.2336163

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "Rao")$Pr[2]
[1] 0.232581
```


Notes:

- In this example, all tests should have been valid because the model was correctly specified.
- This was exemplified (though certainly not proven) by the fact that the resulting p-values were so similar.
- Let's try an example where the model is *not* fully correct, in which case we might expect to see some differences.

TABLE OF CONTENTS

- 1 Hypothesis testing
- 2 Examples: Logistic regression
- 3 Examples: Gaussian regression
- 4 Inverting score tests

Model setup: For simulation

- Exposure: $X \sim \text{Uniform}(1, 10)$.
- Outcome: $Y_i \sim \mathcal{N}(\beta_0 + \beta_1 x_i + \beta_2 x_i^2, \sigma_i^2 = 0.25x_i^2)$.
- Let $\beta_0 = 50$, $\beta_1 = 0$ and $\beta_2 = 0$.
- Let $N = 100$.
- Consider the hypothesis test: $H_0 : \beta_1 = \beta_2 = 0$
 - ▶ No quadratic association between X and mean Y .
- OLS: correctly specified mean model, but model-based variance wrong.
- Consider LR test, score test, Wald test (model-based variance), and Wald test (sandwich-based variance).

Simulated example: Generate data

```
set.seed(7345)
n <- 1000
beta <- matrix(c(50, 0, 0), nrow = 3)
x <- runif(n, 1, 10)
X <- matrix(cbind(1, x, x^2), ncol = 3)
y <- rnorm(n, X %*% beta, x/2)
```

Simulated example: Gaussian GLM

```
## Gaussian GLM
bhat <- solve(t(X) %*% X) %*% t(X) %*% y
phi.hat <- t(y - X %*% bhat) %*% (y - X %*% bhat)/(n - 3)
An <- t(X) %*% X
Bn <- t(X) %*% diag(c(y - X %*% bhat)^2) %*% X
```

Simulated example: Variance estimators (unrestricted)

```
## Model-based variance
V1 <- solve(A_n) * as.numeric(phi.hat)

## Sandwich variance
V2 <- solve(A_n) %*% B_n %*% solve(A_n)
```

Simulated example: Variance estimators (unrestricted)

```
## Restricted model
X0 <- X[,c(-2,-3)]
bhat.0 <- solve(t(X0) %*% X0) %*% t(X0) %*% y
bhat.0 <- cbind(c(bhat.0, 0, 0))
```

Simulated example: Likelihood ratio test

```
## Likelihood ratio test
loglik <- sum(log(dnorm(y, X %*% bhat, sqrt(phi.hat))))
loglik.0 <- sum(log(dnorm(y, X %*% bhat.0, sqrt(phi.hat))))
LR <- -2*(loglik.0 - loglik)
pLR <- 1 - pchisq(LR, df = 2)
```


Simulated example: Score test

```
## Score test
Gn0 <- t(X) %*% (y - X %*% bhat.0)/as.numeric(phi.hat)
An0 <- t(X) %*% X/as.numeric(phi.hat)
S <- t(Gn0) %*% solve(An0) %*% Gn0
pS <- 1 - pchisq(S, df = 2)
```

Simulated example: Wald tests

```
## Wald tests
C <- matrix(0, nrow = 2, ncol = 3)
C[1,2] <- 1
C[2,3] <- 1

## Model-based Wald test
W1 <- t(C %*% bhat) %*% solve(C %*% V1 %*% t(C)) %*% (C %*% bhat)
pW1 <- 1 - pchisq(W1, df = 2)

## Sandwich-based Wald test
W2 <- t(C %*% bhat) %*% solve(C %*% V2 %*% t(C)) %*% (C %*% bhat)
pW2 <- 1 - pchisq(W2, df = 2)
```

Simulated example: Report and compare to ANOVA function

```
> pLR
[1] 0.05750231

> pS
      [,1]
[1,] 0.05750231

> pW1
      [,1]
[1,] 0.05750231

> pW2
      [,1]
[1,] 0.1752994

zz.Full <- glm(y ~ X[,2] + X[,3], family = gaussian)
zz.Reduced <- glm(y ~ 1, family = gaussian)

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "LRT")$Pr[2]
[1] 0.05750231

## Exact match! :)
> anova(zz.Full, zz.Reduced, test = "Rao")$Pr[2]
[1] 0.05750231
```

Discussion questions:

- Which p-values (if any) are valid for the test of interest?
- Can the likelihood ratio and score tests be modified to be valid?

TABLE OF CONTENTS

- 1 Hypothesis testing
- 2 Examples: Logistic regression
- 3 Examples: Gaussian regression
- 4 Inverting score tests

Background:

- The most widely used approach to form a confidence interval for a regression parameter is to use the observed standard error:

$$\widehat{\beta}_k \pm z_{1-\alpha/2} \widehat{SE}[\widehat{\beta}_k].$$

(or, similarly, we could use $t_{df,1-\alpha/2}$)

- This “inverts the Wald test,” in the sense that the confidence interval encompasses all values, c , such that the null hypotheses $H_0 : \beta_k = c$ would not be rejected by a two-sided α -level Wald test.
- What if we wanted to invert, say, the score test instead?
 - ▶ I am able to identify two GLMs in which it is “straightforward enough” to do so: Gaussian and Poisson regression. There may be others.

INVERTING SCORE TESTS

Example: Gaussian regression

- Model: $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon$, $E[\epsilon] = 0$ and $\text{Var}[\epsilon] = \sigma^2$.
- Suppose we use OLS to estimate β (i.e., Gaussian regression with identity link) and seek to form a 95% confidence interval for β_1 by inverting the score test rather than the Wald test.
- Under $H_0 : \beta_1 = c$, the null model is given as:

$$\begin{aligned} Y &= \beta_0 + cX_1 + \beta_2 X_2 \\ \Rightarrow Y - cX_1 &= \beta_0 + \beta_2 X_2. \end{aligned}$$

- This suggests that we can regress $Y - cX_1$ on X_2 to obtain estimates of β_0 and β_2 under H_0 (call the null estimate $\hat{\beta}^0 = (\hat{\beta}_0, c, \hat{\beta}_2)$).
- Compute p-value, p_c associated with score statistic, and search for all c such that $p_c > 0.05$.
- Nice example to illustrate the concept, but “unnecessary” in that Wald and score tests are equivalent for Gaussian regression.

Example: Poisson regression

- Model: $Y \sim \text{Poisson}(\lambda = \exp(\beta_0 + \beta_1 X_1 + \beta_2 X_2))$.
- Suppose we seek to form a 95% confidence interval for β_1 by inverting the score test rather than the Wald test.
- Under $H_0 : \beta_1 = c$, the estimating equations are given as:

$$\begin{aligned}\mathbf{X}^T(\mathbf{y} - \exp(\beta_0 + cX_1 + \beta_2 X_2)) &= \mathbf{0} \\ \Rightarrow \mathbf{X}^T(\mathbf{y} - \exp(cX_1) \exp(\beta_0 + \beta_2 X_2)) &= \mathbf{0}.\end{aligned}$$

- This suggests that we can include cX_1 as an offset in the Poisson model to obtain estimates of β_0 and β_2 under H_0 (call the null estimate $\hat{\boldsymbol{\beta}}^0 = (\hat{\beta}_0, c, \hat{\beta}_2)$).
- Compute p-value, p_c associated with score statistic, and search for all c such that $p_c > 0.05$.

Example: Poisson regression

```
## Set seed for reproducibility
set.seed(7345)

## Sample size of 100
n <- 100

## Generate exposure
X <- runif(n, 1, 5)

## Generate outcome
Y <- rpois(n, lambda = exp(1 + 0.1*X))

## Fit Poisson model
zz <- glm(Y ~ X, family = poisson)

## Wald-based confidence interval
> confint(zz)[2,]
      2.5 %      97.5 %
0.04481821 0.22128675
```

INVERTING SCORE TESTS

Example: Poisson regression

```
## Wald-based confidence interval is a good starting point
> confint(zz)[2,]
      2.5 %      97.5 %
0.04481821 0.22128675

## p-value under offset given by left-hand side
zz <- glm(Y ~ X, offset = 0.04481821*X, family = poisson)
> summary(zz)$coefficients[2,4]
[1] 0.0518257

## p-value under offset given by right-hand side
zz <- glm(Y ~ X, offset = 0.2212868*X, family = poisson)
> summary(zz)$coefficients[2,4]
[1] 0.04794482

## This is from a manual search (not fun), though one could be programmed
zz <- glm(Y ~ X, offset = 0.044126034*X, family = poisson)
> summary(zz)$coefficients[2,4]
[1] 0.05

zz <- glm(Y ~ X, offset = 0.220481791*X, family = poisson)
> summary(zz)$coefficients[2,4]
[1] 0.05

## 95% CI obtained by inverting score test
## [0.044126034, 0.220481791]
```

Further commentary:

- You'll sometimes see confidence intervals formed by inverting other kinds of tests (e.g., permutation tests).
- This can be controversial, especially if the emerging interval is not expected to have proper coverage of a target parameter.
- Some say testing and estimation should be thought of separately.
- Perhaps a compromise would be to refer to these special intervals as “inversion” intervals, and then have a conversation about settings under which an inversion interval and a confidence interval are equivalent or share similar properties.
 - ▶ Recall: there is no reason to believe that a “1/20” likelihood support interval will have 95% coverage.

So far:

- Hypothesis testing for GLMs.

Up next:

- Diagnostics.