

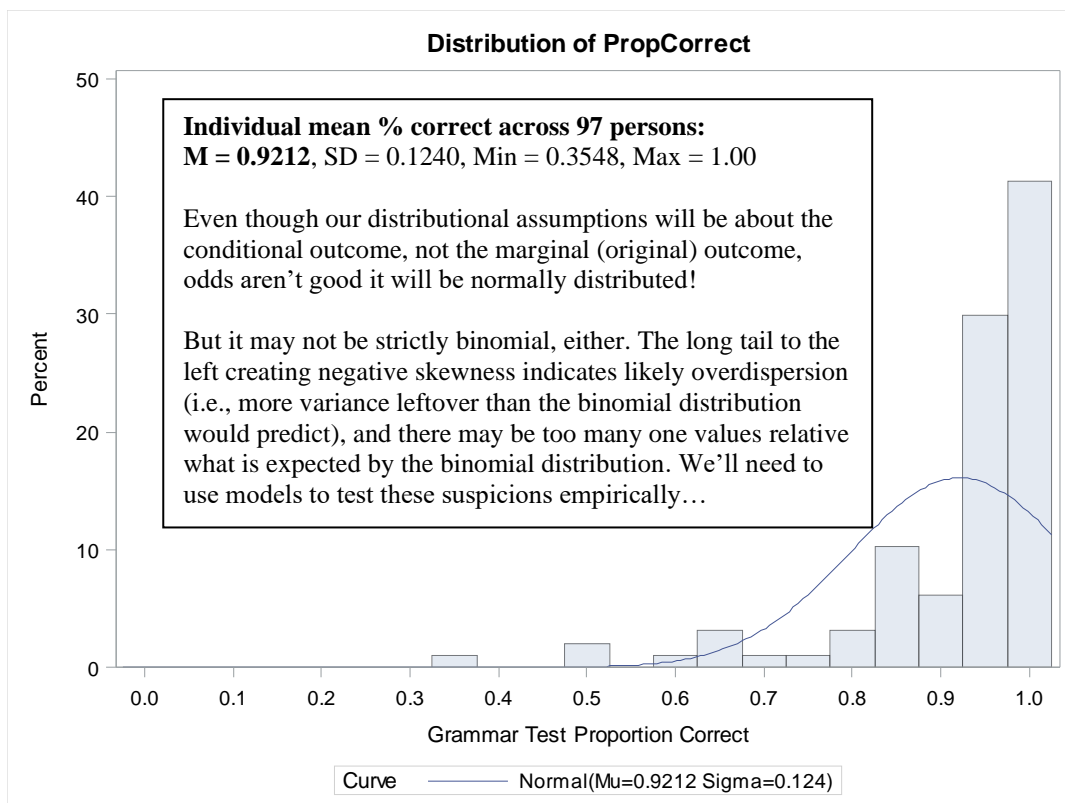
Example 4a: Generalized Linear Models for Binomial Outcomes (% Correct)
(complete syntax and output available for STATA, R, and SAS electronically)

The real data for this example come from the publication below, which examined annual growth in a test of grammatical understanding from Kindergarten through 4th grade in children with non-specific language impairment (NLI) or specific language impairment (SLI):

Rice, M. L., Tomblin, J. B., **Hoffman, L.**, Richman, W. A., & Marquis, J. (2004). [Grammatical tense deficits in children with SLI and nonspecific language impairment: Relationships with nonverbal IQ over time](#). *Journal of Speech-Language-Hearing Research*, 47(4), 816-834.

The current example is a cross-sectional analysis of how grammatical understanding at third grade (measured by proportion correct) is predicted by group (NLI=0, SLI=1) and mother's years of education (centered so that 0=12 years). Given that proportion correct is bounded by 0 and 1, we will use a logit link and the binomial family of conditional response distributions. **Because the binomial is a discrete distribution, we will need to parameterize the model to predict the number of correct responses out of the number of trials directly instead of proportion correct.** This example will also demonstrate two ways of addressing binomial overdispersion: *additive* (through individual random intercepts) and *multiplicative* (through the beta-binomial distribution), as well as zero-inflated (actually one-inflated here; stay tuned) versions of the binomial and beta-binomial model variants (in which the probability of being an extra zero is predicted in a separate submodel using a logit link).

In SAS (as given in online materials only), I am still using GLIMMIX for the binomial models, as well as FMM (finite mixture model) for the beta-binomial and zero-inflated model variants. In STATA, I am using GLM to get conditional distribution fit, as well as MEGLM, BETABIN, ZIB, and ZIBBIN. In R, I am using the VGLM function from the VGAM package and GLMER from the LME4 package. Unfortunately, because the VGLM function uses expected information instead of observed information (as used in STATA and SAS), the standard errors for the parameter estimates (and thus any Wald test results) will differ between STATA/SAS and R. Likelihood ratio test results are the same, however. Btw, in SAS GLIMMIX, I set denominator DF to "none" so that the SAS Wald test results will match those of STATA.



STATA Syntax for Importing and Preparing Data for Analysis:

```
// Defining global variable for file location to be replaced in code below
// \\Client\ precedes path in Virtual Desktop outside H drive;
global filesave "C:\Dropbox\23_PSQF6270\PSQF6270_Example4a"

// Import Example 4a Stata data
use "$filesave\PSQF6270_Example4a.dta", clear

// Label existing variables for analysis
label variable nliivsli      "Group: 0=NLI, 1=SLI"
label variable momed12      "Mother Education (0=12 years)"
label variable propcorrect  "Grammar Test Proportion Correct"
// Create number correct for denominator of binomial outcome
gen ntrials=100
gen ncorrect=round(propcorrect*ntrials,1)
// Compute number incorrect for zero-inflated binomial model
gen nincorrect=ntrials-ncorrect
gen propincorrect=1-propcorrect

// Find betabin and zbin, install before continuing
// search betabin
// search zbin

display "STATA Distribution of Proportion Correct"
summarize propcorrect
hist propcorrect, percent start(0) width(.05)
graph export "$filesave\STATA Proportion Correct Histogram.png", replace
```

R Syntax for Importing and Preparing Data for Analysis (after loading packages *haven*, *TeachingDemos*, *psych*, *multcomp*, *vgam*, and *lme4* as shown online):

```
# Define variables for working directory and data name
filesave = "C:\\Dropbox\\23_PSQF6270\\PSQF6270_Example4a\\"
filename = "PSQF6270_Example4a.sas7bdat"
setwd(dir=filesave)

# Import Example 4a SAS data
Example4a = read_sas(data_file=paste0(filesave,filename))
# Convert to data frame without labels to use for analysis
Example4a = as.data.frame(Example4a)

# Label existing variables for analysis
#NLIvSLI=      "Group: 0=NLI, 1=SLI"
#momed12=      "Mother Education (0=12 years)"
#PropCorrect=  "Grammar Test Proportion Correct"

# Create number correct for denominator of binomial outcome
Example4a$Ntrials=100
Example4a$Ncorrect=Example4a$PropCorrect*Example4a$Ntrials
Example4a$Ncorrect=round(Example4a$Ncorrect,0)
# Compute number incorrect for zero-inflated binomial model
Example4a$Nincorrect=Example4a$Ntrials-Example4a$Ncorrect
Example4a$PropIncorrect=1-Example4a$PropCorrect

print("R Distribution of Proportion Correct")
describe(x=Example4a$PropCorrect)

# To save a plot: open a file, create the plot, then close the file
png(file = "R Proportion Correct Histogram.png") # open file
hist(x=Example4a$PropCorrect, freq=FALSE,
     ylab="Density",xlab="Grammar Test Proportion Correct") # axis labels
dev.off() # close file
```

1) Empty Binomial Model for % correct DV = Events/Trials in SAS and STATA; Events/Non-Events in R

$\#Correct_i \sim \text{Binomial}(p_i, Ntrials_i) \rightarrow p_i$ is probability of any one trial being correct as the “event”

$\text{Logit}(p_i \text{ for correct trial}) = \beta_0$

Conditional mean μ for $\#Correct_i = Ntrials_i * p_i$

Conditional variance for $\#Correct_i = (Ntrials_i * p_i)(1 - p_i)$

STATA Syntax and Partial Output for Empty Binomial Model:

```
display "STATA Empty Binomial Model using glm -- ntrials is denominator"
glm ncorrect, link(logit) family(binomial ntrials) nolog
```

Generalized linear models	No. of obs	=	97	
Optimization : ML	Residual df	=	96	
	Scale parameter	=	1	
Deviance = 1620.05009	(1/df) Deviance	=	16.87552	
Pearson = 2041.435988	(1/df) Pearson	=	21.26496	→ way too high! (1=good)
Variance function: $V(u) = u*(1-u/ntrials)$	[Binomial]			
Link function : $g(u) = \ln(u/(ntrials-u))$	[Logit]			
	AIC	=	19.00196	→ not usual version!
	BIC	=	1180.878	→ not usual version!

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
ncorrect						
_cons	2.459276	.0376936	65.24	0.000	2.385397 2.533154	mean in logits → probability = .9212

To inverse link from logits to predicted % correct:

$$Prob(y = 1) = \frac{\exp(2.4593)}{1 + \exp(2.4593)} = .9212$$

The sample mean probability of getting any item correct is .9212. So even though we are actually modeling number of correct trials as the outcome, the model directly predicts the logit of the **proportion correct** (as the conditional mean p , the probability that any trial = 1).

```
display "-2LL= " e(11)*-2 // Print -2LL for model
-2LL= 1841.1902
```

```
margins // Get intercept in expected # events (ILINK*Ntrials = percent here)
```

	Margin	Std. Err.	z	P> z	[95% Conf. Interval]	
_cons	92.12371	.2735021	336.83	0.000	91.58766 92.65977	mean in # events

R Syntax and Partial Output for Empty Binomial Model:

```
# Save sample size as variable for use in Pearson chi-square/DF fit
N=97
```

```
print("R Empty Binomial Model using vglm and two outcome columns")
ModelEmpty = vglm(data=Example4a, binomialff(link="logitlink", multiple.responses=FALSE),
  formula=cbind(Ncorrect,Nincorrect)~1) # Can also use multiv format of 0/1
print("Print ML -2LL with results"); -2*logLik(ModelEmpty); summary(ModelEmpty)
[1] 1841.1902 → -2LL for model
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	2.459276	0.037693	65.246	< 0.00000000000000022	mean in logits → probability = .9212

Name of linear predictor: logitlink(prob)
Residual deviance: 1620.0501 on 96 degrees of freedom
Log-likelihood: -920.59511 on 96 degrees of freedom

Warning: Hauck-Donner effect detected in the following estimate(s):
'(Intercept)'

```
print("Get intercept in probability")
ModelEmptyProb=1/(1+exp(-1*coefficients(ModelEmpty))); ModelEmptyProb
0.92123711
```

```
print("Pearson Chi-Square / DF Index of Fit")
sum(residuals(ModelEmpty, type="pearson")^2)/(N-1) # STATA N-k
[1] 21.262847 → way too high (1=good)
```

This warning indicates that the intercept had some difficulty being estimated. For more info, see <https://search.r-project.org/CRAN/refmans/VGAM/html/hdeff.html>

2) Two-Predictor Binomial Model

$\#Correct_i \sim Binomial(p_i, Ntrials_i) \rightarrow p_i$ is probability of any one trial being correct as the “event”

$Logit(p_i \text{ for correct trial}) = \beta_0 + \beta_1(NLlvSLI_i) + \beta_2(MotherEd_i - 12)$

Conditional mean μ : $\#Correct_i = Ntrials_i * p_i$

Conditional variance of $\#Correct_i = (Ntrials_i * p_i)(1 - p_i)$

STATA Syntax and Partial Output for Two-Predictor Binomial Model:

```
display "STATA Two-Predictor Binomial Model using glm"
glm ncorrect c.nlivsli c.momed12, link(logit) family(binomial ntrials) nolog
```

Generalized linear models	No. of obs	=	97	
Optimization : ML	Residual df	=	94	
	Scale parameter	=	1	
Deviance = 1310.593044	(1/df) Deviance	=	13.94248	
Pearson = 1448.891028	(1/df) Pearson	=	15.41373	→ still way too high!
Variance function: V(u) = u*(1-u/ntrials)	[Binomial]			
Link function : g(u) = ln(u/(ntrials-u))	[Logit]			
	AIC	=	15.85292	
Log likelihood = -765.8665858	BIC	=	880.5702	

ncorrect	Coef.	OIM Std. Err.	z	P> z	[95% Conf. Interval]		
nlivsli	-1.221578	.0858707	-14.23	0.000	-1.389881	-1.053275	Beta1
momed12	.1193325	.0214268	5.57	0.000	.0773368	.1613283	Beta2
_cons	3.071929	.0746183	41.17	0.000	2.92568	3.218178	Beta0 logit → prob = .956

```
display "-2LL= " e(11)*-2 // Print -2LL for model
-2LL= 1531.7332
```

```
estat ic, n(97) // Print AIC and BIC
```

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
.	97	.	-765.8666	3	1537.733	1545.457

```
test (c.nlivsli=0)(c.momed12=0) // Multiv Wald test of model
chi2( 2) = 273.58 → way too optimistic because distribution fit is bad
Prob > chi2 = 0.0000
```

```
display "STATA Two-Predictor Binomial Model -- get odds ratios using eform"
glm ncorrect c.nlivsli c.momed12, link(logit) family(binomial ntrials) eform
```

ncorrect	Odds Ratio	OIM Std. Err.	z	P> z	[95% Conf. Interval]		
nlivsli	.2947646	.0253116	-14.23	0.000	.2491048	.3487937	exp(Beta1)
momed12	1.126745	.0241425	5.57	0.000	1.080406	1.175071	exp(Beta2)
_cons	21.5835	1.610525	41.17	0.000	18.6469	24.98257	exp(Beta0)

R Syntax and Partial Output for Two-Predictor Binomial Model:

```
print("R Two-Predictor Binomial Model using vglm")
ModelBin = vglm(data=Example4a, binomialff(link="logitlink", multiple.responses=FALSE),
               formula=cbind(Ncorrect,Nincorrect)~1+NLIvSLI+momed12)
print("Print -2LL, AIC, BIC, and results")
-2*logLik(ModelBin); AIC(ModelBin); BIC(ModelBin); summary(ModelBin)
[1] 1531.7332 → -2LL for model
[1] 1537.7332 → AIC
[1] 1545.4573 → BIC

Coefficients:
              Estimate Std. Error  z value      Pr(>|z|)
(Intercept)  3.071929    0.074610  41.1729 < 0.000000000000000022  Beta0 logit → prob = .956
NLIvSLI      -1.221578    0.085864 -14.2269 < 0.000000000000000022  Beta1
momed12      0.119333    0.021426   5.5695  0.00000002555  Beta2

Name of linear predictor: logitlink(prob)
Residual deviance: 1310.593 on 94 degrees of freedom
Log-likelihood: -765.86659 on 94 degrees of freedom

print("Pearson Chi-Square / DF Index of Fit")
sum(residuals(ModelBin, type="pearson")^2)/(N-3) # STATA N-k
[1] 15.410798 → still way too high!

print("Multiv Wald test of model")
BinTestR2 = glht(model=ModelBin, linfct=c("NLIvSLI=0","momed12=0"))
print(summary(BinTestR2, test=Chisqtest()), digits="8") # Joint chi-square test
Global Test:
      Chisq DF      Pr(>Chisq)
1 273.62108  2 3.8365119e-60 → way too optimistic because distribution fit is bad

print("Get odds ratios with 95% CIs")
exp(cbind(OR=coefficients(ModelBin), confint.default(ModelBin)))
              OR          2.5 %          97.5 %
(Intercept) 21.58349665 18.64718757 24.98217632 exp(Beta0)
NLIvSLI      0.29476465  0.24910818  0.34878902 exp(Beta1)
momed12      1.12674455  1.08040748  1.17506895 exp(Beta2)
```

Before interpreting these results, let's see if we can get better distribution fit. Here are some alternative models that incorporate overdispersion, zero-inflation (actually one-inflation here), or both at the same time...

3) Two-Predictor Binomial Model with Additive Overdispersion: Also known as adding an "observation-level random effect" (OLRE)

$\#Correct_i \sim \text{Binomial}(p_i, Ntrials_i) \rightarrow p_i$ is probability of any one trial being correct

$\text{Logit}(p_i \text{ for correct}) = \beta_0 + \beta_1(NLIvSLI_i) + \beta_2(\text{MotherEd}_i - 12) + U_{0i}$

Conditional mean μ of $\#Correct_i = Ntrials_i * p_i$

Conditional variance of $\#Correct_i = (Ntrials_i * p_i)(1 - p_i)$

The random intercept variance is on the model-scale (in logits), and it effectively soaks up all discrepancy to each person's predicted logit.

STATA Syntax and Partial Output for Additive Overdispersion:

```
display "STATA Two-Predictor Binomial Model with Additive Overdispersion"
display "Using meglm instead; || id. adds random intercept variance"
display "Also known as observation-level random effect for overdispersion"
meglm ncorrect c.nlivsli c.momed12, || id: , ///
      link(logit) family(binomial ntrials) intmethod(laplace) nolog
// LRT for added random intercept variance is done for you automatically
// Multiv Wald test of model also given in meglm for you, Pearson chi-square/DF not given
```

```
Mixed-effects GLM                Number of obs    =          97
Family:                          binomial
Link:                             logit
Binomial variable:                ntrials
Group variable:                   id
                                   Number of groups   =          97
                                   Obs per group:

```

```

Integration method:      laplace
min = 1
avg = 1.0
max = 1
Wald chi2(2) = 14.04 → Wald test of model
Prob > chi2 = 0.0009
Log likelihood = -274.88176
-----
ncorrect |      Coef.   Std. Err.    z    P>|z|    [95% Conf. Interval]
-----+-----
nlivsli | -1.793682   .5089051   -3.52  0.000   -2.791118   -.7962467  Beta1 still signif
momed12 |  .0327918   .1341283    0.24  0.807   -2.2300949   .2956784  Beta2 no longer signif
  _cons |  4.742648   .4350784   10.90  0.000    3.88991    5.595386  Beta0 logit → prob=.991
-----+-----
id
var(_cons)|  4.381075   1.028769                2.765034    6.941619  extra variance in logits
-----+-----
LR test vs. logistic model:  chibar2(01) = 981.97      Prob >= chibar2 = 0.0000 → LRT additive overdisp

```

Along with the much larger standard errors (as expected from allowing extra variance), the estimates have also changed because the total model has more variance in it now (as opposed to the total 3.29 residual variance given the logit link).

```

display "-2LL= " e(ll)*-2 // Print -2LL for model
-2LL= 549.76351

```

```

estat ic, n(97) // Print AIC and BIC

```

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
.	97	.	-274.8818	4	557.7635	568.0624

```

display "STATA Two-Predictor Binomial Model with Additive Overdispersion"
display "Using meglm instead and getting odds ratios using eform"
meglm ncorrect c.nlivsli c.momed12, || id: , ///
link(logit) family(binomial ntrials) intmethod(laplace) eform nolog

```

ncorrect	exp(b)	Std. Err.	z	P> z	[95% Conf. Interval]	
nlivsli	.1663465	.0846546	-3.52	0.000	.0613526 .4510186	exp(Beta1) → was 0.29
momed12	1.033335	.1385995	0.24	0.807	.7944582 1.344038	exp(Beta2) → was 1.12
_cons	114.7376	49.91986	10.90	0.000	48.90649 269.1815	exp(Beta0) → was 21.58

R Syntax and Partial Output for Additive Overdispersion:

```

print("R Two-Predictor Binomial Model using glmer")
print("Additive Overdispersion via Random Intercept Variance (1|ID)")
print("Also known as observation-level random effect for overdispersion")
ModelBinAdd = glmer(data=Example4a, family=binomial(link="logit"),
                    formula=cbind(Ncorrect,Nincorrect)~1+NLivSLI+momed12+(1|ID))
print("Print -2LL, AIC, BIC, and results")
-2*logLik(ModelBinAdd); AIC(ModelBinAdd); BIC(ModelBinAdd); summary(ModelBinAdd)
'log Lik.' 549.76432 (df=4) → -2LL for model
[1] 557.76432 → AIC
[1] 568.06316 → BIC

```

```

Random effects:
Groups Name      Variance Std.Dev.
ID (Intercept)  4.3795   2.0927 → extra variance in logits
Number of obs: 97, groups: ID, 97

```

```

Fixed effects:
Estimate Std. Error z value      Pr(>|z|)
(Intercept)  4.741983   0.433979  10.9268 < 0.00000000000000022  Beta0 logit → prob = .991
NLivSLI     -1.793161   0.508270  -3.5280   0.0004188  Beta1 still significant
momed12      0.032825   0.134020   0.2449   0.8065140  Beta2 no longer significant

```

```

print("Pearson Chi-Square / DF Index of Fit")
sum(residuals(ModelBinAdd, type="pearson")^2)/(N-3)
[1] 0.15402372 → much smaller because all extra variance went into random intercept

```

```

print("Multiv Wald test of model")
BinAddTestR2 = glht(model=ModelBinAdd, linfct=c("NLIvSLI=0","momed12=0"))
print(summary(BinAddTestR2, test=Chisqtest()), digits="8") # Joint chi-square test
Global Test:
  Chisq DF      Pr(>Chisq)
1 14.061888  2 0.00088409677 → much smaller test statistic than before

print("Likelihood Ratio Test for Addition of Random Intercept Variance")
DevTestA=-2*(logLik(ModelBin)-logLik(ModelBinAdd))
RegPvalueA=pchisq(DevTestA, df=1, lower.tail=FALSE); MixPvalueA=RegPvalueA/2
print("Test Statistic, Regular and Mixture P-values for DF=1")
DevTestA;          RegPvalueA;          MixPvalueA
'log Lik.' 981.96886 (df=4)  'log Lik.' 1.4914969e-215 (df=4)  'log Lik.' 7.4574845e-216 (df=4)

print("Get odds ratios with 95% CIs")
cbind(OR=exp(ModelBinAdd@beta), exp(confint(ModelBinAdd, parm="beta_", method="Wald")))
      OR      2.5 %      97.5 %
(Intercept) 114.6613778 48.979525220 268.42300959 exp(Beta0) → was 21.58
NLIvSLI      0.1664332  0.061461097  0.45069174 exp(Beta1) → was 0.29
momed12      1.0333697  0.794653011  1.34379771 exp(Beta2) → was 1.12

```

4) Two-Predictor Model with Multiplicative Overdispersion via the Beta-Binomial Distribution

#Correct_i ~ BetaBinomial(p_i, Ntrials_i, φ) → p_i is still probability of any one trial being correct

p_i ~ Beta(a_i, b_i) → a_i = p_i/φ, b_i = (1 - p_i)/φ

Logit(p_i for correct trial) = β₀ + β₁(NLIvSLI_i) + β₂(MotherEd_i - 12)

Conditional mean: #Correct_i = Ntrials_i * p_i

Conditional variance of #Correct_i = (Ntrials_i * p_i)(1 - p_i)[1 + (Ntrials_i - 1)/(φ + 1)]

Disclaimer: I struggled to translate this model across the different parameterizations I found, and this formula for the conditional variance produced results that were close to those software provided but not exactly the same...

STATA Syntax and Partial Output for Multiplicative Overdispersion:

```

display "STATA Two-Predictor Beta-Binomial Model with Multiplicative Overdispersion"
display "Using betabin instead that has beta-binomial distribution"
betabin ncorrect c.nlivsli c.momed12, link(logit) n(ntrials) nolog
// LRT for multiplicative overdispersion is done for you automatically
// Pearson chi-square/DF fit not given

Beta-binomial regression          Number of obs   =          97
Link                             = logit              LR chi2(2)      =       13.61 → LRT for model
Dispersion                       = beta-binomial   Prob > chi2     =       0.0035
Log likelihood = -267.05167       Pseudo R2      =       0.0248

-----+-----
      ncorrect |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-----+-----
      nlivsli |   - .9737565   .2728606    -3.57  0.000   -1.508553   -.4389595  Beta1 still signif
      momed12 |    .0464046   .0685461     0.68  0.498   -.0879434   .1807525  Beta2 now not signif
       _cons |    2.957862   .2500499    11.83  0.000    2.467773    3.44795  Beta0 logit → prob = .951
-----+-----
      /lnsigma |  -1.421521   .2207495    -6.44  0.000   -1.854182   -.9888596  = log(1/phi)
-----+-----
       sigma |    .2413467   .0532772          .156581   .3720007  = 1/phi multiplier in SAS
-----+-----
Likelihood-ratio test of sigma=0:  chibar2(01) =   997.63 Prob>=chibar2 = 0.000 → LRT for overdispersion

display "-2LL= " e(11)*-2 // Print -2LL for model
-2LL= 534.10333

estat ic, n(97) // Print AIC and BIC
-----+-----
      Model |      Obs   ll(null)   ll(model)      df      AIC      BIC
-----+-----
          . |      97 -273.8551  -267.0517         4   542.1033  552.4022
-----+-----

```



```

test (c.nlivsli=0)(c.momed12=0) // Multiv Wald test of model
      chi2( 2) = 14.04
      Prob > chi2 = 0.0009

predict yhatBBpred // Save predicted propcorrect per real person to dataset
corr yhatBBpred propcorrect // Get corr of pred and actual propcorrect
display "R2= " r(rho)^2 // Print R2 relative to empty model
R2= .15730629

display "STATA Two-Predictor Binomial Model with Multiplicative Overdispersion"
display "Using betabin and Getting Odds Ratios using eform"
betabin ncorrect c.nlivsli c.momed12, link(logit) n(ntrials) eform nolog

```

```

-----+-----
ncorrect |      exp(b)   Std. Err.      z    P>|z|      [95% Conf. Interval]
-----+-----
nlivsli |      .3776617   .103049   -3.57   0.000   .2212298   .6447069   exp(Beta1)
momed12 |      1.047498   .0718019   0.68   0.498   .9158127   1.198119   exp(Beta2)
_ cons |      19.25675   4.815148   11.83   0.000   11.79615   31.4359   exp(Beta0)
-----+-----

```

R Syntax and Partial Output for Multiplicative Overdispersion:

```

print("R Two-Predictor Binomial Model using vglm with Multiplicative Overdispersion")
ModelBetaBin = vglm(data=Example4a, betabinomial(lmu="logitlink", lrho="logitlink"),
                    formula=cbind(Ncorrect,Nincorrect)~1+NLivSLI+momed12)
print("Print -2LL, AIC, BIC, and results")
-2*logLik(ModelBetaBin); AIC(ModelBetaBin); BIC(ModelBetaBin); summary(ModelBetaBin)
[1] 534.10333 → -2LL for model
[1] 542.10333 → AIC
[1] 552.40218 → BIC

Coefficients:
      Estimate Std. Error z value      Pr(>|z|)
(Intercept):1  2.957854  0.255237 11.5887 < 0.000000000000000022 Beta0 logit → prob = .951
(Intercept):2 -1.421511  0.214887 -6.6152  0.0000000000003711 Log(1/phi) as given in Stata
NLivSLI        -0.973743  0.271864 -3.5817  0.0003413 Beta1
momed12        0.046404  0.071489  0.6491  0.5162707 Beta2

print("Pearson Chi-Square / DF Index of Fit")
sum(residuals(ModelBetaBin, type="pearson")^2)/(N-3)
[1] 1.7995172 → much closer to 1=good than regular binomial!

print("Multiv Wald test of model")
BinMultTestR2 = glht(model=ModelBetaBin, linfct=c("NLivSLI=0","momed12=0"))
print(summary(BinMultTestR2, test=Chisqtest()), digits="8") # Joint chi-square test
Global Test:
      Chisq DF   Pr(>Chisq)
1 14.86719  2 0.0005910587

print("Likelihood Ratio Test for Addition of Multiplicative Overdispersion")
DevTestB=-2*(logLik(ModelBin)-logLik(ModelBetaBin))
RegPvalueB=pchisq(DevTestB, df=1, lower.tail=FALSE); MixPvalueB=RegPvalueB/2
print("Test Statistic, Regular and Mixture P-values for DF=1")
DevTestB;      RegPvalueB;      MixPvalueB
[1] 997.62984   [1] 5.8810413e-219 [1] 2.9405206e-219

print("Save predicted propcorrect and correlate with actual propcorrect")
Example4a$PredBetaBin = predict(ModelBetaBin, type="response")
rPredBetaBin = cor.test(Example4a$PredBetaBin, Example4a$PropCorrect, method="pearson")
print("R2"); rPredBetaBin$estimate^2
0.15730603

print("Get odds ratios with 95% CIs")
exp(cbind(OR=coefficients(ModelBin), confint.default(ModelBin)))
      OR      2.5 %      97.5 %
(Intercept):1 19.25660777 11.67674520 31.75687544 exp(Beta0)
(Intercept):2 0.24134899 0.15839210 0.36775403 1/phi as given in Stata
NLivSLI        0.37766663 0.22166509 0.64345761 exp(Beta1)
momed12        1.04749703 0.91054552 1.20504687 exp(Beta2)

```


5) Two-Predictor Binomial Model with Zero-Inflation (predicting number incorrect now)

Our negatively skewed data have one-inflation, not zero-inflation, but all the software routines I found were designed only for zero-inflation. So I solved this problem by **predicting number incorrect** instead of number correct. The model below says that number incorrect comes from a binomial distribution that has extra zero values. The “inflation” model that predicts the logit of being an “extra zero” is empty for now, because I just want to see how many extra zeros there are.

$$\text{Logit}(p_{ip} \text{ for incorrect trial}) = \beta_{0p} + \beta_{1p}(NLIvsSLI_i) + \beta_{2p}(MotherEd_i - 12)$$

$$\text{Logit}(p_{iz} \text{ for } y_i = \text{extra } 0) = \beta_{0z}$$

$$\text{Conditional mean: } \#Incorrect_i = (Ntrials_i * p_{ip}) * p_{iz}$$

I'm not even going to try to get the distributional notation or conditional variance right...

STATA Syntax and Partial Output for Binomial with Zero-Inflation (predicting number incorrect):

```
display "STATA Two-Predictor Zero-Inflated Binomial Model"
display "Use zbin and predict nincorrect instead"
display "ilink is for submodel predicting extra 0 (empty here)"
zib nincorrect c.nlivsli c.momed12, link(logit) n(ntrials) ilink(logit) inflate(_cons) nolog
// Pearson chi-square/DF fit not given
```

Zero-inflated binomial regression	Number of obs	=	97	
Regression link: logit	Nonzero obs	=	57	
Inflation link : logit	Zero obs	=	40	
	LR chi2(2)	=	126.58	→ LRT for model
Log likelihood = -494.1091	Prob > chi2	=	0.0000	

nincorrect	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
nincorrect						
nlivsli	.6787023	.0934716	7.26	0.000	.4955014 .8619033	Beta1p
momed12	-.1148639	.024894	-4.61	0.000	-.1636552 -.0660727	Beta2p
_cons	-2.209937	.0825224	-26.78	0.000	-2.371678 -2.048196	Beta0p
inflate						
_cons	-.3547476	.2063317	-1.72	0.086	-.7591502 .049655	Beta0z logit of extra 0 → probability = .4122

```
display "-2LL= " e(11)*-2 // Print -2LL for model
-2LL= 988.2183
```

```
estat ic, n(97) // Print AIC and BIC
```

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
.	97	-557.3991	-494.1091	4	996.2183	1006.517

```
test (c.nlivsli=0)(c.momed12=0) // Multiv Wald test of model
chi2( 2) = 116.04
Prob > chi2 = 0.0000
```

```
display "LRT for zero-inflation (binomial vs zero-inflated binomial)"
```

```
display "Test Statistic (df=1)= " 1531.7332-988.2183
```

```
Test Statistic (df=1)= 543.5149
```

```
display "Regular p-value= " (1-chi2(1, 1531.7332-988.2183))
```

```
0
```

```
display "Mixture p-value= " 0.5*(1-chi2(1, 1531.7332-988.2183))
```

```
0
```

```
display "STATA Two-Predictor Zero-Inflated Binomial Model"
```

```
display "Use zbin and predict nincorrect instead, get odds ratios"
```

```
display "ilink is for submodel predicting extra 0 (empty here)"
```

```
zib nincorrect c.nlivsli c.momed12, link(logit) n(ntrials) ilink(logit) inflate(_cons) eform nolog
```

nincorrect	exp(b)	Std. Err.	z	P> z	[95% Conf. Interval]	
nincorrect						
nlivsli	1.971318	.1842622	7.26	0.000	1.641321 2.367663	exp(Beta1p)
momed12	.8914875	.0221927	-4.61	0.000	.8490347 .9360629	exp(Beta2p)
_cons	.1097075	.0090533	-26.78	0.000	.093324 .1289673	exp(Beta0p)

```
-----+-----
inflate |
  _cons | -.3547476 .2063317 -1.72 0.086 -.7591502 .049655 Beta0z logit of extra 0
-----+-----
                                         → probability = .4122
```

R Syntax and Partial Output for Binomial with Zero-Inflation (predicting number incorrect):

```
print("R Two-Predictor Zero-Inflated Binomial Model using vglm Predicting Nincorrect")
ModelZBin = vglm(data=Example4a, zibinomialff(lprob="logitlink", lonempstr0="logitlink",
      multiple.responses=FALSE, ionempstr0=NULL, zero="onempstr0"),
      formula=cbind(Nincorrect,Ncorrect)~1+NLIvSLI+momed12)
print("Print -2LL, AIC, BIC, and results")
-2*logLik(ModelZBin); AIC(ModelZBin); BIC(ModelZBin); summary(ModelZBin)
[1] 988.2183 → -2LL for model
[1] 996.2183 → AIC
[1] 1006.5171 → BIC

Coefficients:
      Estimate Std. Error z value      Pr(>|z|)
(Intercept):1 -2.209921   0.068295 -32.3583 < 0.000000000000000022 Beta0p
(Intercept):2  0.354748   0.206329  1.7193      0.08555 Beta0z*-1 = logit of not extra 0
NLIvSLI        0.678734   0.084490  8.0333 0.00000000000000009488 Beta1p
momed12       -0.114872   0.022742 -5.0512 0.0000004391152171797 Beta2p

Names of linear predictors: logitlink(prob), logitlink(onempstr0)
Log-likelihood: -494.10915 on 190 degrees of freedom

print("Pearson Chi-Square / DF Index of Fit")
sum(residuals(ModelZBin, type="pearson")^2)/(N-3)
[1] 13.359286 → way too high again!

print("Multiv Wald test of model")
ZBinMultTestR2 = glht(model=ModelZBin, linfct=c("NLIvSLI=0", "momed12=0"))
print(summary(ZBinMultTestR2, test=Chisqtest()), digits="8") # Joint chi-square test
Global Test:
      Chisq DF      Pr(>Chisq)
1 111.55444 2 5.9739459e-25 → way too optimistic given bad distribution fit

print("Likelihood Ratio Test for Addition of Zero-Inflation over Binomial")
DevTestC=-2*(logLik(ModelBin)-logLik(ModelZBin))
RegPvalueC=pchisq(DevTestC, df=1, lower.tail=FALSE); MixPvalueC=RegPvalueC/2
print("Test Statistic, Regular and Mixture P-values for DF=1")
DevTestC;      RegPvalueC;      MixPvalueC
[1] 543.51488 [1] 3.2417789e-120 [1] 1.6208894e-120

print("Get odds ratios with 95% CIs")
exp(cbind(OR=coefficients(ModelZBin), confint.default(ModelZBin)))

      OR      2.5 %      97.5 %
(Intercept):1 0.10970928 0.095964415 0.12542281 exp(Beta0)
(Intercept):2 1.42582081 0.951561912 2.13645058 exp(Beta0z*-1)
NLIvSLI        1.97138047 1.670522825 2.32642193 exp(Beta1)
momed12        0.89148032 0.852617086 0.93211498 exp(Beta2)
```

6) Two-Predictor Beta-Binomial Model with Zero-Inflation (predicting number incorrect)

The model below says that number incorrect comes from a beta-binomial distribution that has extra zero values (instead of a binomial distribution that has extra zero values), allowing multiplicative overdispersion.

$$\text{Logit}(p_{ip} \text{ for incorrect}) = \beta_{0p} + \beta_{1p}(NLIvSLI_i) + \beta_{2p}(MotherEd_i - 12)$$

$$\text{Logit}(p_{iz} \text{ for } y_i = \text{extra } 0) = \beta_{0z}$$

$$\text{Conditional mean: } \#Incorrect_i = (Ntrials_i * p_{ip}) * p_{iz}$$

I'm not even going to try to get the distributional notation or conditional variance right...

I could not find zero-inflated beta-binomial regression in R, so I gave up

STATA Syntax and Partial Output for Beta-Binomial with Zero-Inflation (empty inflation model):

```

display "STATA Two-Predictor Zero-Inflated Beta-Binomial Model"
display "Use zibbin and predict nincorrect instead"
zibbin nincorrect c.nlivsli c.momed12, link(logit) n(ntrials) ilink(logit) inflate(_cons) nolog
// Pearson chi-square/DF fit not given

Zero-inflated beta-binomial regression          Number of obs   =          97
Regression link: logit                        Nonzero obs     =          57
Inflation link : logit                       Zero obs        =          40
                                                LR chi2(2)      =       11.61 → LRT of model
Log likelihood = -263.789                     Prob > chi2     =       0.0030
-----
      nincorrect |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-----+-----
nincorrect      |
  nlivsli       |   1.128224   .3464563     3.26  0.001     .4491825   1.807266  Beta1p
  momed12       |  -.0178967   .0894132    -0.20  0.841    -.1931434   .1573499  Beta2p
  _cons         |  -2.750534   .3270209    -8.41  0.000    -3.391483  -2.109585  Beta0p logit → prob = .06
-----+-----
inflate         |
  _cons         |  -1.095397   .4369649    -2.51  0.012    -1.951832  -.2389614  Beta0z logit of extra 0
                                                → probability = .2506
-----+-----
  /lnsigma      |  -1.870879   .2495082                    -2.359906  -1.381852  → log(1/phi)
-----+-----
  sigma         |   .1539883   .0384213                    .0944291   .2511131  → 1/phi multiplier in SAS
-----+-----
display "-2LL= " e(11)*-2 // Print -2LL for model
-2LL= 527.57802

estat ic, n(97) // Print AIC and BIC
-----+-----
      Model |      Obs   ll(null)   ll(model)      df      AIC      BIC
-----+-----
      .    |      97  -269.5932   -263.789      5     537.578   550.4516
-----+-----

test (c.nlivsli=0)(c.momed12=0) // Multiv Wald test of model
      chi2( 2) = 12.43
      Prob > chi2 = 0.0020

display "LRT for overdispersion (zero-inflated: binomial vs beta-binomial)"
display "Test Statistic (df=1)= " 988.2183-527.5780
Test Statistic (df=1)= 460.6403 → definitely need multiplicative overdispersion
display "Regular p-value= " (1-chi2(1, 988.2183-527.5780))
0
display "Mixture p-value= " 0.5*(1-chi2(1, 988.2183-527.5780))
0

// Save predicted propincorrect per real person to dataset
predict yhatZIBB, xb // Predicted outcome in logits
gen Npred=1/(1+exp(-1*yhatZIBB)) // Convert to probability
corr Npred propincorrect // Get corr of pred and actual propincorrect
display "R2=" r(rho)^2 // Print R2 relative to empty model
R2= .14449233

display "STATA Two-Predictor Zero-Inflated Beta-Binomial Model"
display "Use zibbin and predict nincorrect instead, get odds ratios"
zibbin nincorrect c.nlivsli c.momed12, link(logit) n(ntrials) ilink(logit) inflate(_cons) eform nolog
-----
      nincorrect |      exp(b)   Std. Err.      z    P>|z|     [95% Conf. Interval]
-----+-----
nincorrect      |
  nlivsli       |   3.090165   1.070607     3.26  0.001     1.567031   6.093766  exp(Beta1p)
  momed12       |   .9822625   .0878272    -0.20  0.841     .8243638   1.170405  exp(Beta2p)
  _cons         |   .0638938   .0208946    -8.41  0.000     .0336587   .1212883  exp(Beta0p)
-----+-----
inflate         |
  _cons         |  -1.095397   .4369649    -2.51  0.012    -1.951832  -.2389614  Beta0z logit of extra 0
                                                → probability = .2506
-----+-----
  /lnsigma      |  -1.870879   .2495082                    -2.359906  -1.381852  → log(1/phi)
-----+-----

```

sigma | .1539883 .0384213 .0944291 .2511131 → 1/phi multiplier in SAS

7) Four-Predictor Beta-Binomial Model with Zero-Inflation (now with predictors in inflation model)

This model adds our two predictors to the zero-inflation model (customizing the probability of being an extra zero).

$$\text{Logit}(p_i \text{ for incorrect}) = \beta_{0p} + \beta_{1p}(\text{NLivsSLI}_i) + \beta_{2p}(\text{MotherEd}_i - 12)$$

$$\text{Logit}(p_{iz} \text{ for } y_i = \text{extra } 0) = \beta_{0z} + \beta_{1z}(\text{NLivsSLI}_i) + \beta_{2z}(\text{MotherEd}_i - 12)$$

Conditional mean: $\# \text{Incorrect}_i = (\text{Ntrials}_i * p_i) * p_{iz}$

I'm not even going to try to get the distributional notation or conditional variance right...

STATA Syntax and Partial Output for Beta-Binomial with Zero-Inflation (predictor inflation model):

```
display "STATA Two-Predictor Zero-Inflated Beta-Binomial Model"
display "Switch to zibbin and predict nincorrect instead"
display "Add two predictors of being extra zero"
zibbin nincorrect c.nlivsli c.momed12, link(logit) n(ntrials) ///
      ilink(logit) inflate(c.nlivsli c.momed12) nolog
```

```
Zero-inflated beta-binomial regression          Number of obs   =          97
Regression link: logit                        Nonzero obs     =          57
Inflation link : logit                       Zero obs        =          40
                                                LR chi2(2)      =          7.38
Log likelihood = -261.8274                    Prob > chi2     =          0.0249
```

nincorrect	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]		

nincorrect							
nlivsli	.3036772	.3546852	0.86	0.392	-.391493	.9988474	Beta1p
momed12	-.2189386	.0812336	-2.70	0.007	-.3781535	-.0597237	Beta2p
_cons	-2.173967	.3963158	-5.49	0.000	-2.950731	-1.397202	Beta0p

inflate							→ predict logit of extra 0
nlivsli	-3.970179	5.512301	-0.72	0.471	-14.77409	6.833733	Beta2z
momed12	-.9569979	1.428802	-0.67	0.503	-3.757398	1.843402	Beta1z
_cons	.0198758	.6209887	0.03	0.974	-1.19724	1.236991	Beta0z

/lnsigma	-1.652934	.3139631			-2.26829	-1.037578	→ log(1/phi)

sigma	.1914873	.0601199			.103489	.354312	→ 1/phi multiplier in SAS

```
display "-2LL= " e(11)*-2 // Print -2LL for model
-2LL= 523.65476
```

```
display "STATA Two-Predictor Zero-Inflated Beta-Binomial Model"
display "Switch to zibbin and predict nincorrect instead, get odds ratios"
display "Add two predictors of being extra zero"
zibbin nincorrect c.nlivsli c.momed12, link(logit) n(ntrials) ///
      ilink(logit) inflate(c.nlivsli c.momed12) eform nolog
```

nincorrect	exp(b)	Std. Err.	z	P> z	[95% Conf. Interval]		

nincorrect							
nlivsli	1.354832	.4805387	0.86	0.392	.6760468	2.715151	exp(Beta1p)
momed12	.8033711	.0652607	-2.70	0.007	.6851254	.9420248	exp(Beta2p)
_cons	.1137256	.0450713	-5.49	0.000	.0523014	.2472879	exp(Beta0p)

inflate							
nlivsli	-3.970179	5.512301	-0.72	0.471	-14.77409	6.833733	exp(Beta1z)
momed12	-.9569979	1.428802	-0.67	0.503	-3.757398	1.843402	exp(Beta2z)
_cons	.0198758	.6209887	0.03	0.974	-1.19724	1.236991	exp(Beta0z)

/lnsigma	-1.652934	.3139631			-2.26829	-1.037578	→ log(1/phi)

sigma	.1914873	.0601199			.103489	.354312	→ 1/phi multiplier in SAS

So which one should be pick? Let's do some informal model comparisons using distribution fit and relative fit (* may not be exactly comparable due to differences in estimation technique, but they should be close)

Two-Predictor Model	Pearson Chi-Square / DF	-2LL*	AIC*	BIC*
2. Regular Binomial	15.41	1531.73	1537.73	1545.46
3. +Additive Overdispersion	0.15	549.76	557.76	568.10
4. Beta-Binomial (multiplicative)	1.80	534.10	542.10	552.40
5. Zero-Inflated Binomial	13.36	988.22	996.22	1006.52
6. Zero-Inflated Beta-Binomial	(SAS) 0.87	527.58	537.58	550.45
7. ZIBB + Predictors	?	523.65	537.65	555.68

Sample results using all programs (final model = zero-inflated beta-binomial without inflation predictors):

The extent that grammatical understanding (measured either as percent correct or percent incorrect; see below) at third grade could be predicted by language impairment group (non-specific=0, specific=1) and mother's years of education (centered such that 0=12 years) was examined in a series of generalized linear models. In the sample of $N = 97$ children, the mean percent correct was 0.92, with a large percentage of observations at or near the ceiling (1.00). Accordingly, we predicted the number of correct trials out of the number of possible trials using a logit link function to keep the predicted proportion correct outcomes bounded at 1. The type of model specifies that the number of correct responses follows a binomial-based distribution with 100 total trials and the model predicts the logit (log-odds) of a correct answer for any trial. Predicted outcomes in a logit metric can be translated into proportion correct via an inverse link function (which provides model-predicted proportions and their standard errors). All models were estimated using maximum likelihood within SAS GLIMMIX and FMM to assess distribution fit (or Stata glm, betabin, zib, and zibbin, or R vglm and glmer); predictor fixed effects were tested univariately using z-distributions without denominator degrees of freedom. Effect sizes are provided below as odds ratios: the exponentiated logit coefficient in which values from 0 to 1 indicate negative associations, 1 indicates no association, and values above 1 indicate positive associations.

Before interpreting our results, we tested the fit of models with alternative binomial-based conditional outcome distributions (each with main effects of group and mother's education) by examining the Pearson χ^2/DF statistic (in which 1=good fit), as well as likelihood ratio tests (i.e., treating -2 times the difference in log-likelihood between nested models as a χ^2 statistic with degrees of freedom equal to the number of additional parameters). As expected given the negatively skewed observed distribution, a model specifying a standard binomial distribution for number correct did not fit well (Pearson $\chi^2/DF = 15.41$). Two methods of allowing overdispersion were then examined. First, we allowed additive overdispersion via an observation-level random intercept, which significantly improved model fit, $-2\Delta LL(1) = 987.97$, $p < .0001$, but created a tendency towards underdispersion (Pearson $\chi^2/DF = 0.15$). Second, we allowed multiplicative overdispersion by using a beta-binomial distribution, which significantly improved model fit, $-2\Delta LL(1) = 997.63$, $p < .0001$, and appeared to fit well (Pearson $\chi^2/DF = 1.80$). We then examined the potential for one-inflation by predicting number *incorrect* instead so that zero-inflation models could be fitted. A model predicting number incorrect with a zero-inflated binomial distribution was examined but did not fit as well (Pearson $\chi^2/DF = 13.36$), although using a zero-inflated beta-binomial distribution instead did result in good fit (Pearson $\chi^2/DF = 0.87$), as well as the lowest AIC and BIC of all the models. We also examined group and mother's education as predictors of zero-inflation but neither was significant (with higher AIC and BIC values), and thus the empty (unconditional) zero-inflation model was retained.

The model results indicated that 25.06% of the sample were predicted to be an extra 0 (i.e., to be part of the zero-inflated component of the distribution for number incorrect). Otherwise, the predicted intercept for a child with non-specific language impairment whose mother had 12 years of education was a logit = -2.75 , which translates into percent incorrect = 0.06. Children with specific language impairment were predicted to have significantly more incorrect responses (logit = 1.12, OR = 3.09), although no significant difference was found for mother's years of education (logit = -0.02 , OR = 0.98). The scale parameter for multiplicative overdispersion was $1/0.15$, which was significant, $-2\Delta LL(1) = 460.64$, $p < .0001$, relative to the zero-inflated binomial alternative.