

# *Generalized Linear Models*

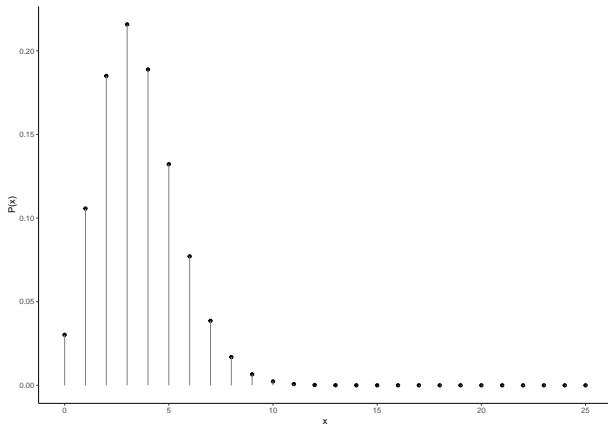
Mark Andrews

Psychology Department, Nottingham Trent University

✉ `mark.andrews@ntu.ac.uk`

# *The Poisson Distribution*

- The Poisson distribution is a discrete probability distribution over the non-negative integers  $0, 1, 2, \dots$



Shown here is a Poisson distribution with  $\lambda = 3.5$ .

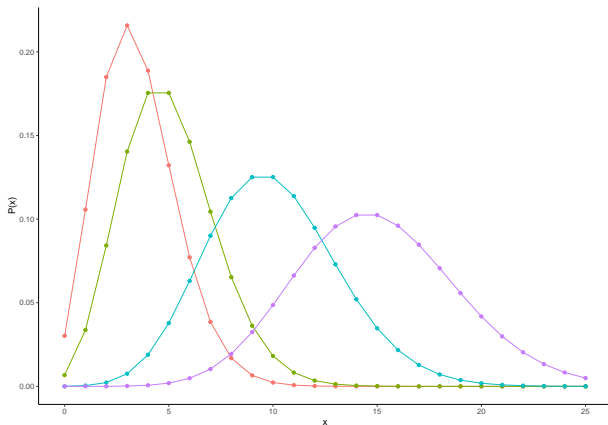
# *The Poisson Distribution*

- ▶ The Poisson distribution is used to model the probability of a given number of events occurring in a fixed interval of time, e.g. the number of emails you get per hour, the number of shark attacks on Bondi beach every summer, etc.
- ▶ It has a single parameter  $\lambda$ , known as the *rate*.
- ▶ If  $x$  is a Poisson random variable whose, its probability mass function is

$$P(x = k|\lambda) = \frac{e^{-\lambda}\lambda^k}{k!}.$$

# *The Poisson Distribution*

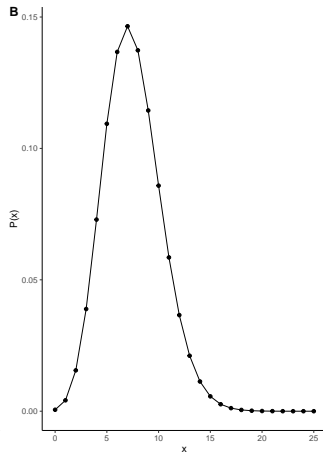
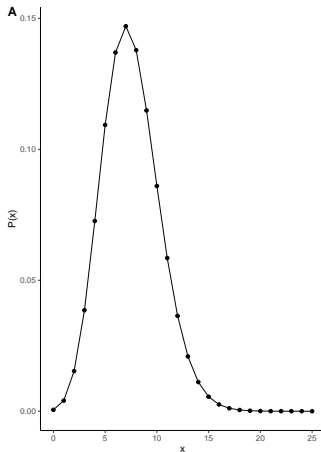
- The mean of a Poisson distribution is equal to its rate parameter  $\lambda$ .
- Its variance is also equal to  $\lambda$ .



As  $\lambda$  increases, so too does the variance.

# The Poisson Distribution

- The Poisson distribution can be seen as the limit of a Binomial distribution as  $N \rightarrow \infty$ , and  $\lambda = pN$ .
- Shown are (left) Binomial( $N, p = \lambda/N$ ) where  $N \approx 10^3$  and  $\lambda = 7.5$ , and (right) Poisson( $\lambda$ ).



# Poisson Regression

- In any regression problem, our data are  $(y_1, x_1), (y_2, x_2) \dots (y_n, x_n)$ , where each  $y_i$  is modelled as a stochastic function of  $x_i$ .
- In Poisson regression, we assume that each  $y_i$  is a Poisson random variable rate  $\lambda_i$  and

$$\log(\lambda_i) = \beta_0 + \sum_{k=1}^K \beta_k x_{ki},$$

or equivalently

$$\lambda_i = e^{\beta_0 + \sum_{k=1}^K \beta_k x_{ki}}.$$

## *Poisson Regression*

- ▶ As an example of Poisson regression, we can look at the number visits to a doctor in a fixed period as a function of predictors such as gender.
- ▶ Using a data-set of over 5000 people, we estimate (using mle) that

$$\log(\lambda_i) = 1.65 + 0.43 \times x_i$$

where  $x_i = 1$  for a female, and  $x_i = 0$  for a male.

## *Poisson Regression*

- Using this example, we see that for a female

$$\lambda_{\text{Female}} = e^{1.65+0.43} = 8.004$$

and for males

$$\lambda_{\text{Male}} = e^{1.65} = 5.2$$

- In other words, the expected value for females is 8.2 and for males it is 5.2.



## Coefficients

- ▶ In Poisson regression, coefficients can be understood as follows.
- ▶ In the previous example,

$$\begin{aligned}\lambda_{\text{Female}} &= e^{1.65+0.43}, \\ &= e^{1.65} e^{0.43}, \\ \lambda_{\text{Male}} &= e^{1.65}.\end{aligned}$$

- ▶ This means that the exponent of the gender coefficient, i.e.  $e^{0.43}$ , signifies the multiplicative increase to the average rate of doctor visits for women relative men.
- ▶ In other words, women visit doctors on average  $e^{0.43} = 1.53$  times more than men.

# Coefficients

- In an arbitrary example with a single continuous predictor variable,

$$\begin{aligned}\lambda &= e^{\alpha + \beta x_i}, \\ &= e^{\alpha} e^{\beta x_i},\end{aligned}$$

If we increase  $x_i$  by 1, we have

$$\begin{aligned}\lambda^+ &= e^{\alpha + \beta (x_i + 1)}, \\ &= e^{\alpha + \beta x_i + \beta}, \\ &= e^{\alpha} e^{\beta x_i} e^{\beta},\end{aligned}$$

- As  $\lambda^+ = \lambda e^{\beta}$ , we see that  $e^{\beta}$  is the multiplicative effect of an increase in one unit to the predictor variable.

## Example

```
doc_df <- read_csv(here('data/DoctorAUS.csv')) %>%  
  mutate(gender = ifelse(sex == 1, 'female', 'male'))  
  
M <- glm(doctorco ~ gender,  
        data = doc_df,  
        family = poisson)  
  
doc_df_new <- tibble(gender = c('female', 'male'))  
  
doc_df_new %>%  
  add_predictions(M)  
  
doc_df_new %>%  
  add_predictions(M, type='response')
```

## *Model comparison*

```
M_1 <- glm(doctorco ~ gender + insurance,  
           data = doc_df,  
           family = poisson)  
  
anova(M, M_1, test='Chisq')
```

## *Exposure and offset*

- ▶ In some problems, the length of time during which events are measured varies across individuals.
- ▶ In the doctor visits example, we might have recordings of number of visits per year for some people and number of visits per 9 months, etc, for others.
- ▶ These situations are dealt with using an *exposure* term for each individual.

## *Exposure and offset*

- ▶ When using an exposure term, we use the original count data as before, but treat

$$y_i \sim \text{Poisson}(\lambda_i/u_i),$$

where  $u_i$  is a term signifying the relative exposure time for person  $i$ .

- ▶ According to this,

$$\log(\lambda_i/u_i) = \alpha + \beta x_i,$$

$$\log(\lambda_i) = \alpha + \beta x_i + \log(u_i)$$

- ▶ In other words,  $y_i \sim \text{Poisson}(\lambda_i/u_i)$  is equivalent to  $y_i \sim \text{Poisson}(\lambda_i)$ , where  $\log(\lambda_i) = \alpha + \beta x_i + \log(u_i)$ .

## *Exposure and offset*

- ▶ For example, suppose we monitor people's drinking at social occasions. We find that three people drink 12, 7 and 3 drinks over the course of 7, 5 and 2 hours, respectively.
- ▶ If we are trying to predict drinking as a function of predictor variables, we ought to calibrate by the different time frames.
- ▶ Treating e.g. 12 as a draw from  $\text{Poisson}(\lambda_i/7)$  where  $\log(\lambda_i/7) = \alpha + \beta x_i$  is identical to treating 12 as a draw from  $\text{Poisson}(\lambda_i)$  where  $\log(\lambda_i) = \alpha + \beta x_i + \log(7)$ .

## *Exposure and offset*

- ▶ In general, exposure terms are treated as fixed offsets.
- ▶ If our data is  $(y_1, x_1), (y_2, x_2) \dots (y_n, x_n)$  with exposures  $u_1, u_2 \dots u_n$ , then we treat

$$y_i \sim \text{Poisson}(\lambda_i),$$

where

$$\log(\lambda_i) = \log(u_i) + \beta_0 + \sum_{k=1}^K \beta_k x_{ki}.$$

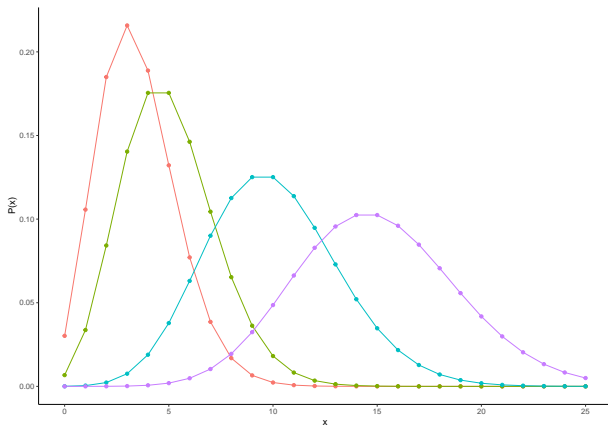


## Example

```
insur_df <- read_csv(here('data/Insurance.csv')) %>%  
  mutate(District = factor(District))  
  
M <- glm(Claims ~ District + Group + Age + offset(log(Holders)),  
  data = insur_df,  
  family = poisson)
```

# *The Poisson Distribution*

- The mean of a Poisson distribution is equal to its rate parameter  $\lambda$ .
- Its variance is also equal to  $\lambda$ .



As  $\lambda$  increases, so too does the variance.

## *Means and variances in a Poisson distribution:*

- ▶ In a Poisson distribution, the variance of a sample should be approximately the same as the mean of a sample.
- ▶ Example 1:

```
x <- rpois(25, lambda = 5)
c(mean(x), var(x), var(x)/mean(x))
#> [1] 5.4400000 5.4233333 0.9969363
```

- ▶ Example 2:

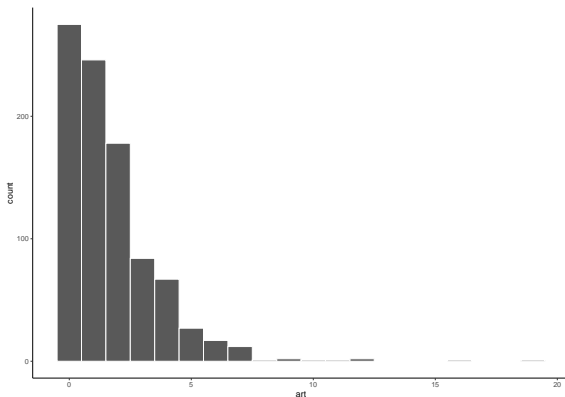
```
x <- rpois(25, lambda = 5)
c(mean(x), var(x), var(x)/mean(x))
#> [1] 5.4000000 6.416667 1.188272
```

# Overdispersion

- ▶ If the variance of a sample is greater than would be expected according to a given theoretical model, then we say the data is *overdispersed*.
- ▶ In count data, if the variance of a sample is much greater than its mean, we say it is overdispersed.
- ▶ Using a Poisson distribution in this situation, this is an example of model mis-specification.
- ▶ It will also usually underestimate the standard errors in the Poisson model.

# Overdispersion

- In the bioChemists data set, we have counts of the number of articles published by PhD students in the last three years (publications):



```
var(publications)/mean(publications)
#> [1] 2.191358
```

# Overdispersion

- This leads standard errors to be *underestimated* if we use a Poisson model:

```
M <- glm(publications ~ 1, family=poisson)
summary(M)$coefficients
```

#>	Estimate	Std. Error	z value	Pr(> z )
#> (Intercept)	0.5264408	0.02540804	20.71945	2.312911e-95

## Fixing overdispersion using a Quasi-poisson model

- ▶ A *quasi* Poisson model allows us to correct over-dispersion

```
M <- glm(publications ~ 1, family=quasipoisson)
summary(M)$coefficients
```

#>	Estimate	Std. Error	t value	Pr(> t )
#> (Intercept)	0.5264408	0.03761239	13.99647	1.791686e-40

- ▶ It does so by calculating an overdispersion parameter (roughly, the ratio of the variance to the mean) and multiplying the standard error by its square root.
- ▶ In this example, the overdispersion parameter is 2.1913892 and so its square root is 1.4803341.
- ▶ Alternatively, a *negative binomial regression* is an alternative to Poisson regression that can be used with overdispersed count data.

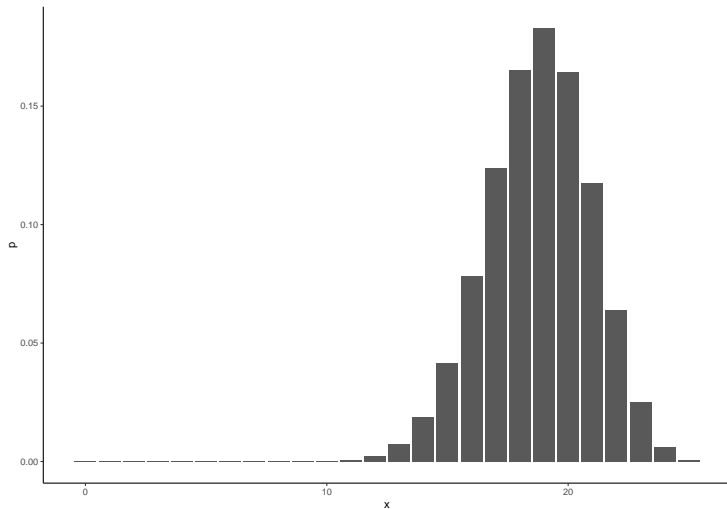
## Negative binomial distribution

- ▶ A negative binomial distribution is a distribution over non-negative integers.
- ▶ To understand the negative binomial distribution, we start with the binomial distribution:
- ▶ If, for example, we have a coin whose probability of coming up heads is  $\theta$ , then the number of Heads in a sequence of  $n$  flips will follow a binomial distribution.
- ▶ In this example, an outcome of Heads can be termed a *success*.



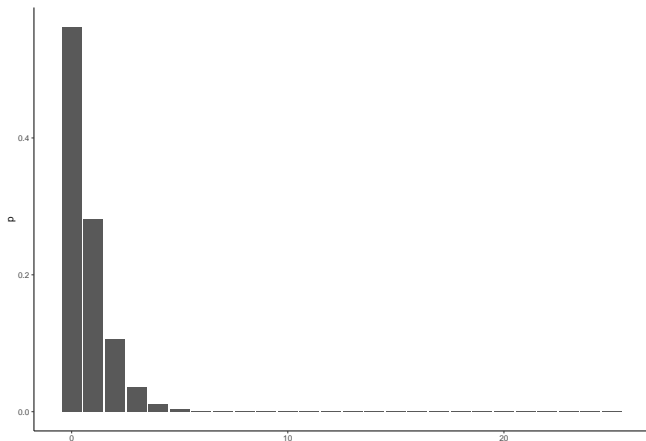
## *Negative binomial distribution*

- Here is a binomial distribution where  $n = 25$  and  $\theta = 0.75$ .



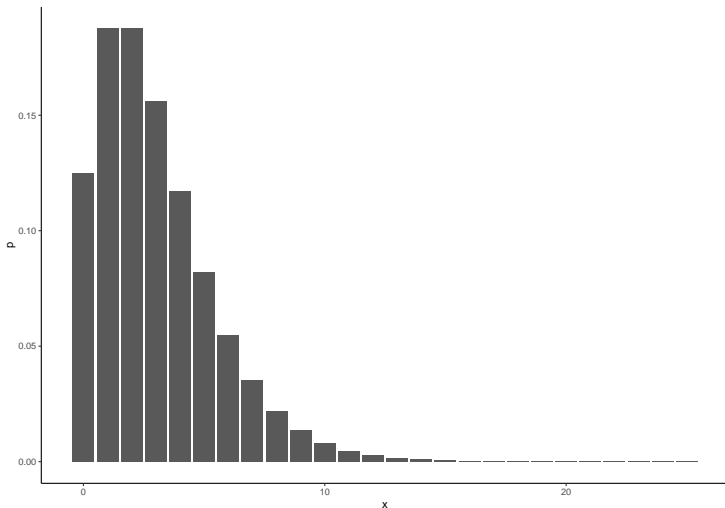
## Negative binomial distribution

- ▶ A *negative* binomial distribution gives the probability distribution over the number of *failures* (e.g. Tails) before  $r$  *successes* (e.g.  $r$  Heads).
- ▶ For example, here we have the number of Tails (*failures*) that occur before we observe  $r = 2$  Heads (*successes*), when the probability of Heads is  $\theta = 0.75$ :



## Negative binomial distribution

- Here, we have the number of Tails (*failures*) that occur before we observe  $r = 3$  Heads (*successes*), when the probability of Heads is  $\theta = 0.5$ :



## Negative binomial distribution

- The probability mass function for the negative binomial distribution is:

$$P(x = k|r, \theta) = \binom{r+k-1}{k} \theta^r (1-\theta)^k$$

or more generally

$$P(x = k|r, \theta) = \frac{\Gamma(r+k)}{\Gamma(r)k!} \theta^r (1-\theta)^k,$$

where  $\Gamma()$  is a Gamma function ( $\Gamma(n) = (n-1)!$ ).

- In R, for any  $k$ ,  $r$ , and  $\theta$ , we can calculate  $P(x = k|r, \theta)$  using `dnbinom`, e.g.  $P(x = k = 2|r = 3, \theta = 0.75)$  is

```
dnbinom(2, 3, 0.75)  
#> [1] 0.1582031
```

## *Negative binomial distribution*

- In the negative binomial distribution, the mean is

$$\mu = \frac{\theta}{1 - \theta} \times r,$$

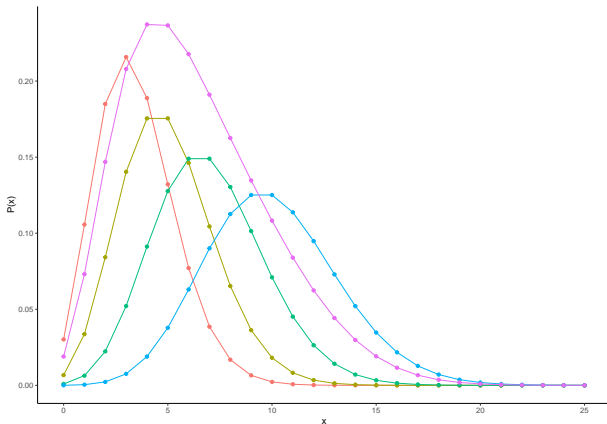
and so

$$\theta = \frac{r}{r + \mu},$$

and we can generally parameterize the distribution by  $\mu$  and  $r$ .

## Why use negative binomial distribution?

- A negative binomial distribution is equivalent as weighted sum of Poissons.



- So it is appropriate to use when the data can be seen as arising from a mixture of Poisson distributions, each with different means.

## *Negative binomial regression*

- In negative binomial regression, we have observed counts  $y_1, y_2 \dots y_n$ , and some predictor variables  $x_1, x_2 \dots x_n$ , and we assume that

$$y_i \sim \text{NegBinomial}(\mu_i, r),$$

where  $\text{NegBinomial}(\mu_i, r)$  is a negative binomial with mean  $\mu_i$  and a dispersion parameter  $r$ , and then

$$\log(\mu_i) = \beta_0 + \beta x_i.$$

## Example

```
M <- glm.nb(publications ~ gender, data = biochemists_Df)
M1 <- glm.nb(publications ~ gender + married + I(children > 0),
```



## *Binomial logistic regression*

- If  $y_i$  is the number of “successes” in  $n_i$  “trials”, we can model this as

$$y_i \sim \text{Binomial}(\theta_i, n_i),$$

$$\text{logit}(\theta_i) = \beta_0 + \sum_{k=1}^K \beta_k x_{ki}$$

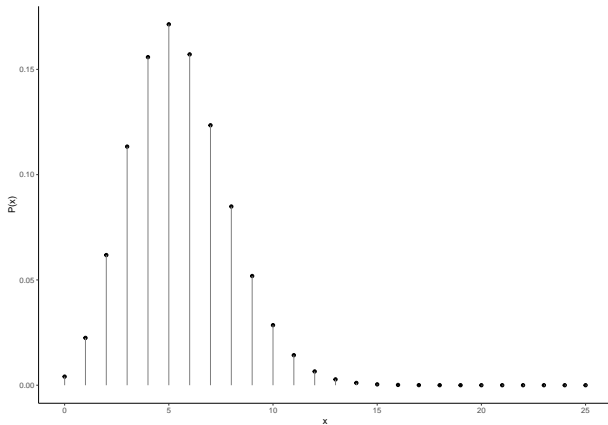
- If  $n_i = 1$  for all  $i$ , then this is exactly binary logistic regression.
- In general, it models the probability of something happening in a number of independent trials, and how the probability varies by the values of predictors.

## Example

```
golf_df <- read_csv(here('data/golf_putts.csv')) %>%  
  mutate(failure = attempts - success,  
         p = success/attempts)  
  
M <- glm(cbind(success, failure) ~ distance,  
        family = binomial(link = 'logit'),  
        data = golf_df)
```

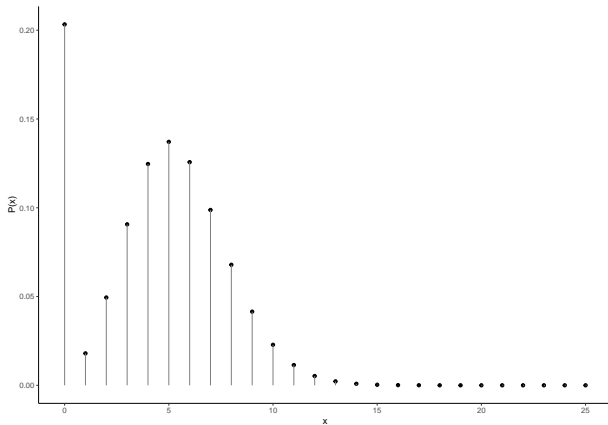
# Poisson Distribution

A sample from a Poisson distribution with  $\lambda = 5.5$ .



# Zero inflated Poisson Distribution

A sample from a zero inflated Poisson distribution with  $\lambda = 5.5$ , with probability of zero-component is 0.2.



## *Poisson regression to Zero-Inflated Poisson regression*

- In Poisson regression (with a single predictor, for simplicity), we assume that each  $y_i$  is a Poisson random variable with rate  $\lambda_i$  that is a function of the predictor  $x_i$ .
- In Zero-Inflated Poisson regression, we assume that each  $y_i$  is distributed as a Zero-Inflated Poisson mixture model:

$$y_i \sim \begin{cases} \text{Poisson}(\lambda_i) & \text{if } z_i = 0, \\ 0, & \text{if } z_i = 1 \end{cases}$$

where rate  $\lambda_i$  and  $P(z_i = 1)$  are functions of the predictor  $x_i$ .

## *Zero-Inflated Poisson regression*

- Assuming data  $\{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}$ , Poisson regression models this data as:

$$y_i \sim \begin{cases} \text{Poisson}(\lambda_i) & \text{if } z_i = 0, \\ 0, & \text{if } z_i = 1 \end{cases},$$
$$z_i \sim \text{Bernoulli}(\theta_i),$$

where  $\theta_i$  and  $\lambda_i$  are functions of  $x_i$ .

## *Zero-Inflated Poisson regression*

- The  $\theta_i$  and  $\lambda_i$  variables are the usual suspects, i.e.

$$\log(\lambda_i) = \alpha + \beta x_i,$$

and

$$\log\left(\frac{\theta_i}{1-\theta_i}\right) = a + bx_i.$$

- In other words,  $\lambda_i$  is modelled just as in ordinary Poisson regression and  $\theta_i$  is modelled in logistic regression.

## Examples

```
smoking_df <- read_csv(here('data/smoking.csv'))
M <- glm(cigs ~ educ, data = smoking_df)
M_zip <- zeroinfl(cigs ~ educ, data=smoking_df)

Df_new <- data.frame(educ = seq(20))
# Predicted average smoking rate
Df_new %>%
  add_predictions(M_zip, type='response')

# Predicted average smoking rate of "smokers"
Df_new %>%
  add_predictions(M_zip, type='count')

# Predicted probability of being a "smoker"
Df_new %>%
  add_predictions(M_zip, type='zero') %>%
  mutate(pred = 1-pred)
```