

Регрессионный анализ для бинарных данных

Линейные модели...

Вадим Хайтов, Марина Варфоломеева



Мы рассмотрим

- ▶ Регрессионный анализ для бинарных зависимых переменных

Вы сможете

- ▶ Построить логистическую регрессионную модель, подобранную методом максимального правдоподобия
- ▶ Дать трактовку параметрам логистической регрессионной модели
- ▶ Провести анализ девиансы, основанный на логистической регрессии

Бинарные данные - очень распространенный тип зависимых переменных

- ▶ Вид есть - вида нет
- ▶ Кто-то в результате эксперимента выжил или умер
- ▶ Пойманное животное заражено паразитами или здорово
- ▶ Команда выиграла или проиграла

и т.д.



На каком острове лучше искать ящериц?



```
liz <- read.csv("data/polis.csv")  
head(liz)
```

#	X.ISLAND	PARATIO	UTA	PA	PREDICT
# 1	Bota	15.41	P	1	0.555
# 2	Cabeza	5.63	P	1	0.915
# 3	Cerraja	25.92	P	1	0.111
# 4	Coronadito	15.17	A	0	0.568
# 5	Flecha	13.04	P	1	0.678
# 6	Gemelose	18.85	A	0	0.370

Зависит ли встречаемость ящериц от размера острова?

Обычную линейную регрессию подобрать можно,

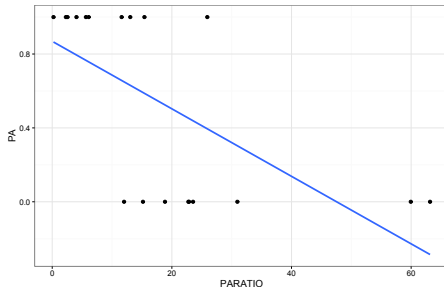
Зависимая переменная:

PA - (есть ящерицы "1" - нет ящериц "0")

Предиктор:

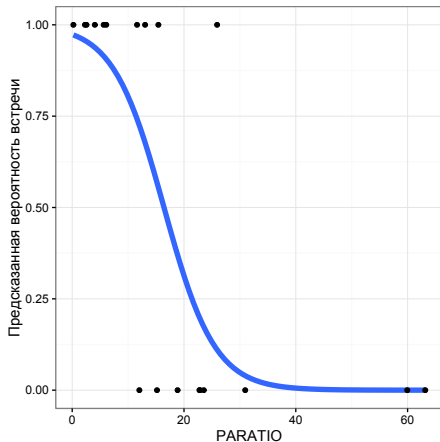
PARATIO (отношение периметра к площади)

```
fit <- lm(PA ~ PARATIO, data = liz)
summary(fit)
```



но она категорически не годится

Эти данные лучше описывает логистическая кривая



Логистическая кривая описывается такой формулой

$$\pi(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$$

Зависимую величину можно преобразовать в более удобную для моделирования форму

1. Дискретный результат: 1 или 0



Зависимую величину можно преобразовать в более удобную для моделирования форму

1. Дискретный результат: 1 или 0
2. Дискретные данные можно преобразовать в форму оценки вероятности события: $\pi = \frac{N_i}{N_{total}}$, непрерывная величина, варьирующая от 0 до 1

Зависимую величину можно преобразовать в более удобную для моделирования форму

1. Дискретный результат: 1 или 0
2. Дискретные данные можно преобразовать в форму оценки вероятности события: $\pi = \frac{N_i}{N_{total}}$, непрерывная величина, варьирующая от 0 до 1
3. Вероятность события можно выразить в форме шансов (odds): $odds = \frac{\pi}{1-\pi}$ варьируют от 0 до $+\infty$. NB: Если шансы > 1 , то вероятность события, что $y_i = 1$ выше, чем вероятность события $y_i = 0$. Если шансы < 1 , то наоборот. В обыденной речи мы часто используем фразы, наподобие такой "шансы на победу 1 к 3"

Зависимую величину можно преобразовать в более удобную для моделирования форму

1. Дискретный результат: 1 или 0
2. Дискретные данные можно преобразовать в форму оценки вероятности события: $\pi = \frac{N_i}{N_{total}}$, непрерывная величина, варьирующая от 0 до 1
3. Вероятность события можно выразить в форме шансов (odds): $odds = \frac{\pi}{1-\pi}$ варьируют от 0 до $+\infty$. NB: Если шансы > 1 , то вероятность события, что $y_i = 1$ выше, чем вероятность события $y_i = 0$. Если шансы < 1 , то наоборот. В обыденной речи мы часто используем фразы, наподобие такой "шансы на победу 1 к 3"
4. Шансы преобразуются в Логиты (logit): $\ln(odds) = \ln\left(\frac{\pi}{1-\pi}\right)$ варьируют от $-\infty$ до $+\infty$. Логиты гораздо удобнее для построения моделей.

Что станет с логистической моделью после логит-преобразования?

Немного алгебры

Обозначим для краткости $\beta_0 + \beta_1 x \equiv z$

Тогда

$$g(x) = \ln\left(\frac{\pi(x)}{1 - \pi(x)}\right) = \ln\left(\frac{\frac{e^z}{1+e^z}}{1 - \frac{e^z}{1+e^z}}\right)$$

Что станет с логистической моделью после логит-преобразования?

Немного алгебры

Обозначим для краткости $\beta_0 + \beta_1 x \equiv z$

Тогда

$$g(x) = \ln\left(\frac{\pi(x)}{1 - \pi(x)}\right) = \ln\left(\frac{\frac{e^z}{1+e^z}}{1 - \frac{e^z}{1+e^z}}\right)$$

$$g(x) = \ln\left(\frac{e^z}{1 + e^z}\right) - \ln\left(1 - \frac{e^z}{1 + e^z}\right)$$

Что станет с логистической моделью после логит-преобразования?

Немного алгебры

Обозначим для краткости $\beta_0 + \beta_1 x \equiv z$

Тогда

$$g(x) = \ln\left(\frac{\pi(x)}{1 - \pi(x)}\right) = \ln\left(\frac{\frac{e^z}{1+e^z}}{1 - \frac{e^z}{1+e^z}}\right)$$

$$g(x) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(1 - \frac{e^z}{1+e^z}\right)$$

$$g(x) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(\frac{1+e^z - e^z}{1+e^z}\right) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(\frac{1}{1+e^z}\right)$$



Что станет с логистической моделью после логит-преобразования?

Немного алгебры

Обозначим для краткости $\beta_0 + \beta_1 x \equiv z$

Тогда

$$g(x) = \ln\left(\frac{\pi(x)}{1 - \pi(x)}\right) = \ln\left(\frac{\frac{e^z}{1+e^z}}{1 - \frac{e^z}{1+e^z}}\right)$$

$$g(x) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(1 - \frac{e^z}{1+e^z}\right)$$

$$g(x) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(\frac{1+e^z - e^z}{1+e^z}\right) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(\frac{1}{1+e^z}\right)$$

$$g(x) = \ln(e^z) - \ln(1+e^z) - (\ln(1) - \ln(1+e^z))$$

Что станет с логистической моделью после логит-преобразования?

Немного алгебры

Обозначим для краткости $\beta_0 + \beta_1 x \equiv z$

Тогда

$$g(x) = \ln\left(\frac{\pi(x)}{1 - \pi(x)}\right) = \ln\left(\frac{\frac{e^z}{1+e^z}}{1 - \frac{e^z}{1+e^z}}\right)$$

$$g(x) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(1 - \frac{e^z}{1+e^z}\right)$$

$$g(x) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(\frac{1+e^z - e^z}{1+e^z}\right) = \ln\left(\frac{e^z}{1+e^z}\right) - \ln\left(\frac{1}{1+e^z}\right)$$

$$g(x) = \ln(e^z) - \ln(1+e^z) - (\ln(1) - \ln(1+e^z))$$

$$g(x) = \ln(e^z) - \ln(1+e^z) - 0 + \ln(1+e^z) = \ln(e^z) = z$$



Логистическая модель после логит-преобразования становится линейной

$$g(x) = \ln\left(\frac{\pi(x)}{1 - \pi(x)}\right) = \beta_0 + \beta_1 x$$

Остается только подобрать параметры этой линейной модели: β_0 (интерсепт) и β_1 (угловой коэффициент)



Метод максимального правдоподобия

Вспомним

Если остатки не подчиняется нормальному распределению, то метод наименьших квадратов не работает.

В этом случае применяют *Метод максимального правдоподобия*

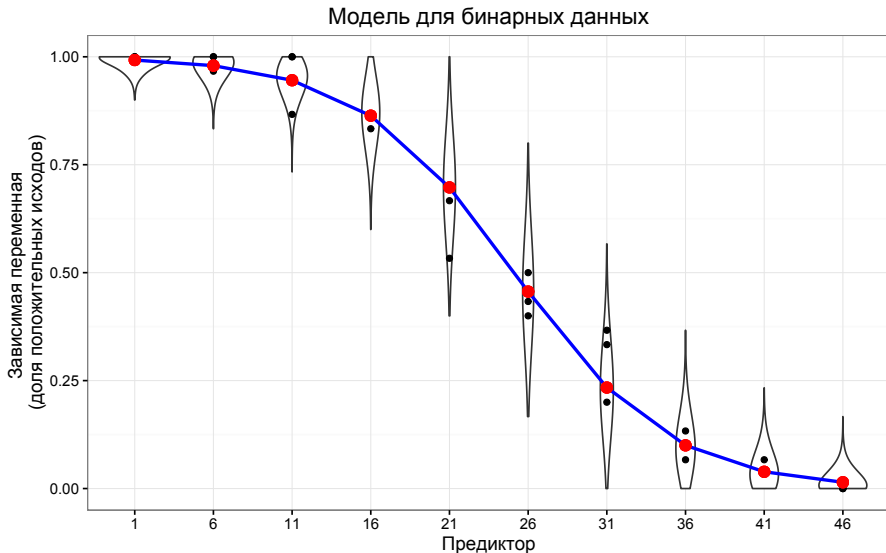
В результате итеративных процедур происходит подбор таких значений коэффициентов, при которых правдоподобие - вероятность получения имеющегося у нас набора данных - оказывается максимальным, при условии справедливости данной модели.

$$Lik(x_1, \dots, x_n) = \prod_{i=1}^n f(x_i; \theta)$$

где $f(x; \theta)$ - функция плотности вероятности с параметрами θ



Правдоподобие для биномиального распределения



Функция правдоподобия для биномиального распределения

Для случая биномиального распределения $x \in \text{Bin}(n, \pi)$ функция правдоподобия имеет следующий вид:

$$\text{Lik}(\pi|x) = \frac{n!}{(n-x)!x!} \pi^x (1-\pi)^{n-x}$$

Отбросив константу, получаем:

$$\text{Lik}(\pi|x) \propto \pi^x (1-\pi)^{n-x}$$

Логарифм правдоподобия

Удобнее работать с логарифмом функции правдоподобия - $\log\text{Lik}$ - его легче максимизировать. В случае биномиального распределения он выглядит так:

$$\log\text{Lik}(\pi|x) = x\log(\pi) + (n-x)\log(1-\pi)$$



Подберем модель

```
liz_model <- glm(PA ~ PARATIO , family="binomial", data = liz)
summary(liz_model)
```

```
#
# Call:
# glm(formula = PA ~ PARATIO, family = "binomial", data = liz)
#
# Deviance Residuals:
#      Min       1Q   Median       3Q      Max
# -1.607  -0.638   0.237   0.433   2.099
#
# Coefficients:
#              Estimate Std. Error z value Pr(>|z|)
# (Intercept)    3.606     1.695    2.13   0.033 *
# PARATIO       -0.220     0.101   -2.18   0.029 *
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# (Dispersion parameter for binomial family taken to be 1)
#
#      Null deviance: 26.287  on 18  degrees of freedom
# Residual deviance: 14.221  on 17  degrees of freedom
# AIC: 18.22
#
```



summary() для модели, подобранной методом максимального правдоподобия

```
#
# Call:
# glm(formula = PA ~ PARATIO, family = "binomial", data = liz)
#
# Deviance Residuals:
#      Min       1Q   Median       3Q      Max
# -1.607  -0.638   0.237   0.433   2.099
#
# Coefficients:
#              Estimate Std. Error z value Pr(>|z|)
# (Intercept)    3.606      1.695    2.13   0.033 *
# PARATIO        -0.220      0.101   -2.18   0.029 *
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# (Dispersion parameter for binomial family taken to be 1)
#
#      Null deviance: 26.287  on 18  degrees of freedom
# Residual deviance: 14.221  on 17  degrees of freedom
# AIC: 18.22
#
# Number of Fisher Scoring iterations: 6
```



"z value" и "Pr(>z)"

z - это величина критерия Вальда (*Wald statistic*) - аналог t-критерия

Используется для проверки $H_0 : \beta_1 = 0$

$$z = \frac{\beta_1}{SE_{\beta_1}}$$

Сравнивают со стандартным нормальным распределением (z-распределение)

Дает надежные оценки p-value при больших выборках

Null deviance и Residual deviance

Имеющиеся данные позволяют “вписать” три типа моделей

“Насыщенная” модель - модель, подразумевающая, что каждая из n точек имеет свой собственный параметр, следовательно надо подобрать n параметров. Вероятность существования данных для такой модели равна 1.

$$\log Lik_{satur} = 0$$

$$df_{saturated} = n - npar_{saturated} = n - n = 0$$

“Нулевая” модель - модель, подразумевающая, что для описания всех точек надо подобрать только 1 параметр. $g(x) = \beta_0$.

$$\log Lik_{nul} \neq 0$$

$$df_{null} = n - npar_{null} = n - 1$$

“Предложенная” модель - модель, подобранная в нашем анализе
 $g(x) = \beta_0 + \beta_1 x$

$$\log Lik_{prop} \neq 0$$

$$df_{proposed} = n - npar_{proposed}$$



Null deviance и Residual deviance

Девianza - это оценка отклонения логарифма максимального правдоподобия одной модели от логарифма максимального правдоподобия другой модели

Остаточная девианса:

$$Dev_{resid} = 2(\log Lik_{satur} - \log Lik_{prop}) = -2\log Lik_{prop}$$

Нулевая девианса:

$$Dev_{nul} = 2(\log Lik_{satur} - \log Lik_{nul}) = -2\log Lik_{nul}$$

Проверим, совпадут ли со значениями из `summary()`

```
(Dev_resid <- -2*as.numeric(logLik(liz_model))) #Остаточная девианса
```

```
# [1] 14.2
```

```
(Dev_nul <- -2*as.numeric(logLik(update(liz_model, ~-PARATIO)))) #Нулевая девианса
```

```
# [1] 26.3
```



Анализ девиансы

По соотношению нулевой девиансы и остаточной девиансы можно понять насколько статистически значима модель

В основе анализа девиансы лежит критерий G^2

$$G^2 = -2(\log Lik_{nul} - \log Lik_{prop})$$

```
(G2 <- Dev_nul - Dev_resid)
```

```
# [1] 12.1
```

Вспомним тест отношения правдоподобий:

$$LRT = 2\ln(Lik_1/Lik_2) = 2(\log Lik_1 - \log Lik_2)$$

Тест G^2 - это частный случай теста отношения правдоподобий (Likelihood Ratio Test)



Свойства критерия G^2

- ▶ G^2 - это девианса полной (предложенной) и редуцированной модели (нулевой)

Свойства критерия G^2

- ▶ G^2 - это девианса полной (предложенной) и редуцированной модели (нулевой)
- ▶ G^2 - аналог частного F критерия в обычном регрессионном анализе

Свойства критерия G^2

- ▶ G^2 - это девианса полной (предложенной) и редуцированной модели (нулевой)
- ▶ G^2 - аналог частного F критерия в обычном регрессионном анализе
- ▶ G^2 - подчиняется χ^2 распределению (с параметом $df = 1$) если нулевая модель и предложенная модель не отличаются друг от друга.

Свойства критерия G^2

- ▶ G^2 - это девианса полной (предложенной) и редуцированной модели (нулевой)
- ▶ G^2 - аналог частного F критерия в обычном регрессионном анализе
- ▶ G^2 - подчиняется χ^2 распределению (с параметом $df = 1$) если нулевая модель и предложенная модель не отличаются друг от друга.
- ▶ G^2 можно использовать для проверки гипотезы о равенстве нулевой и остаточной девианс.

Задание

1. Вычислите вручную значение критерия G^2 для модели, описывающей встречаемость ящериц (`liz_model`)
2. Оцените уровень значимости для него

$$G^2 = -2(\log Lik_{nul} - \log Lik_{prop})$$

Решение

#Остаточная девианса

```
Dev_resid <- -2*as.numeric(logLik(liz_model))
```

#Нулевая девианса

```
Dev_nul <- -2*as.numeric(logLik(update(liz_model, ~-PARATIO)))
```

Значение критерия

```
(G2 <- Dev_nul - Dev_resid)
```

```
# [1] 12.1
```

```
(p_value <- 1 - pchisq(G2, df = 1))
```

```
# [1] 0.000513
```

Решение с помощью функции `anova()`

```
anova(liz_model, test="Chi")
```

```
# Analysis of Deviance Table
#
# Model: binomial, link: logit
#
# Response: PA
#
# Terms added sequentially (first to last)
#
#
```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
# NULL			18	26.3	
# PARATIO 1	1	12.1	17	14.2	0.00051 ***
# ---					

```
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



Интерпретация коэффициентов логистической регрессии



Как трактовать коэффициенты подобранной модели?

$$g(x) = \ln\left(\frac{\pi(x)}{1 - \pi(x)}\right) = \beta_0 + \beta_1 x$$

```
coef(liz_model)
```

# (Intercept)	PARATIO
# 3.61	-0.22

β_0 - не имеет особого смысла, просто поправочный коэффициент

β_1 - *на сколько* единиц изменяется логарифм величины шансов (odds), если значение предиктора изменяется на единицу

Трактовать такую величину неудобно и трудно

посмотрим как изменится $g(x) = \ln\left(\frac{\pi(x)}{1-\pi(x)}\right)$ при изменении предиктора на 1

$$g(x+1) - g(x) = \ln(odds_{x+1}) - \ln(odds_x) = \ln\left(\frac{odds_{x+1}}{odds_x}\right)$$

Задание: завершите алгебраическое преобразование

$$\ln\left(\frac{odds_{x+1}}{odds_x}\right) = \beta_0 + \beta_1(x+1) - \beta_0 - \beta_1x = \beta_1$$

$$\ln\left(\frac{odds_{x+1}}{odds_x}\right) = \beta_1$$

$$\frac{odds_{x+1}}{odds_x} = e^{\beta_1}$$

Полученная величина имеет определенный смысл

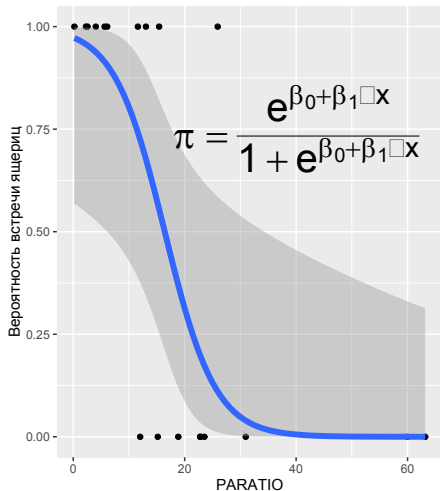
```
exp(coef(liz_model)[2])
```

```
# PARATIO  
# 0.803
```

Во сколько раз изменяются шансы встретить ящерицу при увеличении отношения периметра острова к его площади на одну единицу. NB: Отношение периметра к площади тем больше, чем меньше остров.

Шансы изменяются в 0.803 раза. То есть, чем больше отношение периметра к площади, тем меньше шансов встретить ящерицу. Значит, чем больше остров, тем больше шансов встретить ящерицу

Подобранные коэффициенты позволяют построить логистическую кривую



Серая область - доверительный интервал для логистической регрессии
Доверительные интервалы для коэффициентов:

```
confint(liz_model) # для ЛОГИТОВ
```

#	2.5 %	97.5 %
# (Intercept)	1.006	8.0421
# PARATIO	-0.485	-0.0665

```
exp(confint(liz_model)) # для ОТНОШЕНИЙ
```

#	2.5 %	97.5 %
# (Intercept)	2.734	3109.275
# PARATIO	0.616	0.936

Задание:

Постройте график логистической регрессии для модели `liz_model` без использования `geom_smooth()`

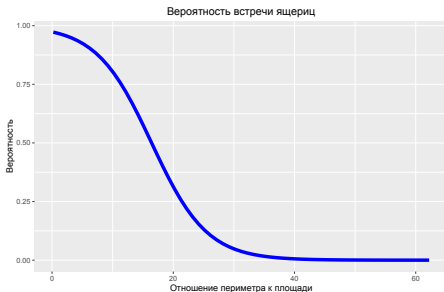
Hint 1: Используйте функцию `predict()`, изучите значения параметра `"type"`

Hint 2: Для вызова справки напишите `predict.glm()`

Hint 3: Создайте датафрейм `MyData` с переменной `PARATIO`, изменяющейся от минимального до максимального значения `PARATIO`



Решение



```
MyData <- data.frame(PARATIO =  
  seq(min(liz$PARATIO), max(liz$PARATIO))
```

```
MyData$Predicted <- predict(liz_model  
  newdata = MyData,  
  type = "response")
```

```
ggplot(MyData, aes(x = PARATIO, y = Pre  
  geom_line(size=2, color = "blue") +  
  xlab("Отношение периметра к площади") +  
  ylab("Вероятность") +  
  ggtitle("Вероятность встречи ящериц"))
```


Применим матричную алгебру для вычисления предсказанных значений и доверительного интервала для линии регрессии

```
# Создаем искусственный набор данных  
MyData <- data.frame(PARATIO = seq(min(liz$PARATIO), max(liz$PARATIO)))  
  
# Формируем модельную матрицу для искусственно созданных данных  
X <- model.matrix( ~ PARATIO, data = MyData)
```

Извлекаем характеристики подобранной модели и получаем предсказанные значения

```
# Вычисляем параметры подобранной модели и ее матрицу ковариаций  
betas      <- coef(liz_model) # Вектор коэффициентов  
Covbetas <- vcov(liz_model) # Ковариационная матрица  
  
# Вычисляем предсказанные значения, перемножая модельную матрицу на вектор  
# коэффициентов  
MyData$eta <- X %*% betas
```

Получаем предсказанные значения

```
# Переводим предсказанные значения из логитов в вероятности  
MyData$Pi <- exp(MyData$eta) / (1 + exp(MyData$eta))
```



Вычисляем границы доверительного интервала

Вычисляем стандартные отшибки путем перемножения матриц

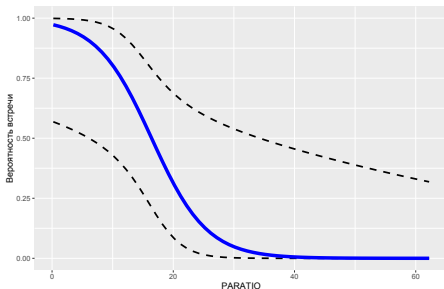
```
MyData$se <- sqrt(diag(X %*% Covbetas %*% t(X)))
```

Вычисляем доверительные интервалы

```
MyData$CiUp <- exp(MyData$eta + 1.96 *MyData$se) /  
(1 + exp(MyData$eta + 1.96 *MyData$se))
```

```
MyData$CiLow <- exp(MyData$eta - 1.96 *MyData$se) /  
(1 + exp(MyData$eta - 1.96 *MyData$se))
```

Строим график



```
ggplot(MyData, aes(x = PARATIO, y = Pi))  
  geom_line(aes(x = PARATIO, y = CiUp),  
            linetype = 2, size = 1) +  
  geom_line(aes(x = PARATIO, y = CiLow),  
            linetype = 2, size = 1) +  
  geom_line(color = "blue", size=2) +  
  ylab("Вероятность встречи")
```

Множественная логистическая регрессия

От чего зависит уровень смертности пациентов, выписанных из реанимации?

Данные, полученные на основе изучения 200 историй болезни пациентов одного из американских госпиталей

- ▶ STA: Статус (0 = Выжил, 1 = умер)
- ▶ AGE: Возраст
- ▶ SEX: Пол
- ▶ RACE: Раса
- ▶ SER: Тип мероприятий в реанимации (0 = Medical, 1 = Surgical)
- ▶ CAN: Присутствует ли онкология? (0 = No, 1 = Yes)
- ▶ CRN: Присутствует ли почечная недостаточность (0 = No, 1 = Yes)
- ▶ INF: Возможность инфекции (0 = No, 1 = Yes)
- ▶ CPR: CPR prior to ICU admission (0 = No, 1 = Yes)
- ▶ SYS: Давление во время поступления в реанимацию (in mm Hg)
- ▶ HRA: Пульс (beats/min)
- PRE: Была ли госпитализация в предыдущие 6 месяцев (0 = No, 1 = Yes) - TYP: Тип госпитализации (0 = Elective, 1 = Emergency) - FRA: Присутствие переломов (0 = No, 1 = Yes) - PO2: Концентрация кислорода в крови (0 = >60 , 1 = ≤ 60) - PH: Уровень кислотности крови (0 = ≤ 7.25 , 1 = > 7.25) - PCO: Концентрация углекислого газа в крови (0 = ≤ 45 , 1 = > 45) - BIC: Bicarbonate from initial blood gases (0 = ≤ 18 , 1 = > 18) - CRE: Уровень креатина (0 = ≤ 2.0 , 1 = > 2.0) - LOC: Уровень сознания пациента при реанимации (0 = no coma or stupor, 1 = deep stupor, 2 = coma)



Смотрим на данные

```
surviv <- read.table("data/ICU.csv", header=TRUE, sep=";")  
head(surviv)
```

#	STA	AGE	SEX	RAC	SER	CAN	CRN	INF	CPR	SYS	HRA	PRE	TYP
# 1	0	27	Female	White	Medical	No	No	Yes	No	142	88	No	Emergency
# 2	0	59	Male	White	Medical	No	No	No	No	112	80	Yes	Emergency
# 3	0	77	Male	White	Surgical	No	No	No	No	100	70	No	Elective
# 4	0	54	Male	White	Medical	No	No	Yes	No	142	103	No	Emergency
# 5	0	87	Female	White	Surgical	No	No	Yes	No	110	154	Yes	Emergency
# 6	0	69	Male	White	Medical	No	No	Yes	No	110	132	No	Emergency
#	FRA	P02	PH	PCO	BIC	CRE	LOC						
# 1	No	1	1	1	1	1	1						
# 2	No	1	1	1	1	1	1						
# 3	No	1	1	1	1	1	1						
# 4	Yes	1	1	1	1	1	1						
# 5	No	1	1	1	1	1	1						
# 6	No	2	1	1	2	1	1						

Сделаем факторами те дискретные предикторы, которые обозначены цифрами

```
surviv$P02 <- factor(surviv$P02)
surviv$PH <- factor(surviv$PH)
surviv$PC0 <- factor(surviv$PC0)
surviv$BIC <- factor(surviv$BIC)
surviv$CRE <- factor(surviv$CRE)
surviv$LOC <- factor(surviv$LOC)
```



Строим модель

```
M1 <- glm(STA ~ ., family = "binomial", data = surviv)
summary(M1)
```

```
#
# Call:
# glm(formula = STA ~ ., family = "binomial", data = surviv)
#
# Deviance Residuals:
#      Min       1Q   Median       3Q      Max
# -1.5052  -0.5372  -0.1787  -0.0002   3.0171
#
# Coefficients:
#              Estimate Std. Error z value Pr(>|z|)
# (Intercept)  -22.44426  1314.00889  -0.02   0.9864
# AGE           0.05645   0.01848   3.05   0.0023 **
# SEXMale       0.72146   0.54600   1.32   0.1864
# RACOther     16.75746  1314.00721   0.01   0.9898
# RACWhite     16.17455  1314.00665   0.01   0.9902
# SERSurgical  -0.67386   0.62894  -1.07   0.2840
# CANYes       3.48260   1.12114   3.11   0.0019 **
# CRNYes       0.11914   0.84488   0.14   0.8879
# INFYes      -0.10812   0.55570  -0.19   0.8457
# CPRYes       1.03223   0.99008   1.04   0.2971
# CVC         0.02084   0.00044   2.21   0.0272 *
```



Задание

Проведите анализ девиансы для данной модели



Решение

```
anova(M1, test = "Chi")
```

```
# Analysis of Deviance Table
#
# Model: binomial, link: logit
#
# Response: STA
#
# Terms added sequentially (first to last)
#
#
```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
# NULL			199	200	
# AGE	1	7.9	198	192	0.00507 **
# SEX	1	0.0	197	192	0.97572
# RAC	2	1.3	195	191	0.53364
# SER	1	8.3	194	183	0.00392 **
# CAN	1	0.7	193	182	0.39726
# CRN	1	2.6	192	179	0.10559
# INF	1	2.0	191	177	0.15391
# CPR	1	3.9	190	173	0.04777 *
# SYS	1	5.7	189	168	0.01741 *
# HRA	1	0.8	188	167	0.37537
# RPE	1	0.0	187	166	0.22055



Упростим модель с помощью функции `step()`

```
step(M1, direction = "backward")
```

Эта функция автоматически применяет функцию `drop1()`, пошагово отбрасывая избыточные предикторы.

Рассмотрим финальную модель

```
M2 <- glm(formula = STA ~ AGE + CAN + SYS + TYP + PH + PCO + LOC, family = "b")  
  
# M2 вложена в M1 следовательно их можно сравнить тестом отношения правдоподобия  
anova(M1, M2, test = "Chi")
```

```
# Analysis of Deviance Table
```

```
#
```

```
# Model 1: STA ~ AGE + SEX + RAC + SER + CAN + CRN + INF + CPR + SYS + HRA +  
# PRE + TYP + FRA + P02 + PH + PCO + BIC + CRE + LOC
```

```
# Model 2: STA ~ AGE + CAN + SYS + TYP + PH + PCO + LOC
```

```
# Resid. Df Resid. Dev Df Deviance Pr(>Chi)
```

```
# 1 178 112
```

```
# 2 191 123 -13 -11.1 0.6
```



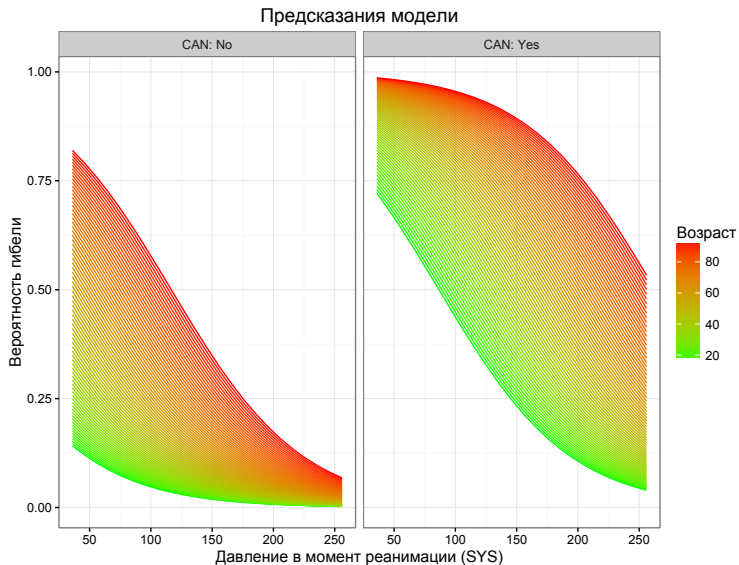
Во сколько раз изменяется отношение шансов на выживание при условии, что пациент онкологический больной (при прочих равных условиях)?

```
exp(coef(M2)[3])
```

```
# CANYes
```

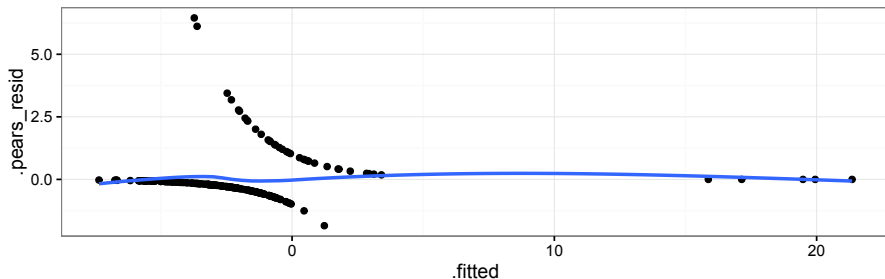
```
# 15.7
```


Визуализируем предсказания модели



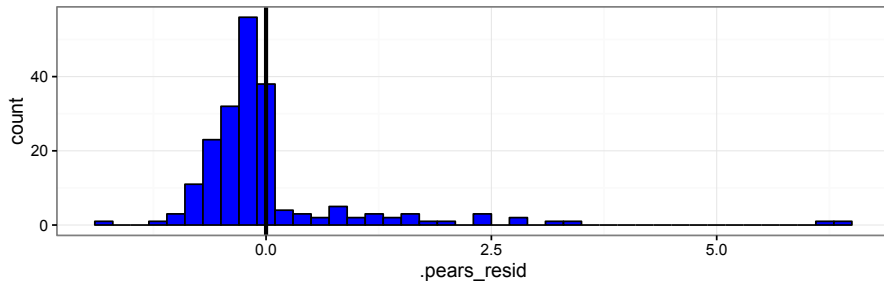
Диагностика модели

```
M2_diag <- data.frame(.fitted = predict(M2),  
  .pears_resid = residuals(M2, type = "pearson"))  
  
ggplot(M2_diag, aes(x = .fitted, y = .pears_resid)) +  
  geom_point() + geom_smooth(se = FALSE)
```



Явного паттерна в остатках нет, но есть другая проблема

Zero inflation



Преобладают отрицательные остатки. Это связано с проблемой, называемой *"zero inflation"*, - в зависимой переменной слишком много нулей.

Сколько должно быть нулей?

#Формируем искусственный набор данных

```
MyData = expand.grid(  
  AGE = seq(min(surviv$AGE),  
            max(surviv$AGE), 1),  
  CAN = levels(surviv$CAN),  
  SYS = seq(min(surviv$SYS),  
            max(surviv$SYS), 10),  
  TYP = levels(surviv$TYP),  
  PH = levels(surviv$PH),  
  PC0 = levels(surviv$PC0),  
  LOC = levels(surviv$LOC)  
)
```

Предсказываем для этих данных вероятности

гибели в соответствии с моделью M2

```
Predicted <- predict(M2, newdata = MyData, type = "response")
```

Вычисляем долю нулей, ожидаемую

в соответствии с биномиальным

распределением

```
Zero_perc <- sum((1-Predicted)) / (sum((1-Predicted)) + sum((Predicted))) * 1
```



Сколько должно быть нулей?

Нулей должно быть 41 %.

А в наших данных доля нулей составляет 80 %.

Это больше, чем должно быть в соответствии с биномиальным распределением.

Нужна более сложная модель!

- ▶ При построении модели для бинарной зависимой переменной применяется логистическая регрессия.

Summary

- ▶ При построении модели для бинарной зависимой переменной применяется логистическая регрессия.
- ▶ При построении такой модели 1 и 0 в переменной отклика заменяются логитами.

Summary

- ▶ При построении модели для бинарной зависимой перменной применяется логистическая регрессия.
- ▶ При построении такой модели 1 и 0 в перменной отклика заменяются логитами.
- ▶ Угловые коэффициенты подобранной логистической регрессии говорят о том, во сколько раз изменяется соотношение шансов события при увеличении предиктора на единицу.

Summary

- ▶ При построении модели для бинарной зависимой перменной применяется логистическая регрессия.
- ▶ При построении такой модели 1 и 0 в перменной отклика заменяются логитами.
- ▶ Угловые коэффициенты подобранной логистической регрессии говорят о том, во сколько раз изменяется соотношение шансов события при увеличении предиктора на единицу.
- ▶ Оценить статистическую значимость модели можно с помощью анализа девиансы.

- ▶ Кабаков Р.И. R в действии. Анализ и визуализация данных на языке R. М.: ДМК Пресс, 2014.
- ▶ Quinn G.P., Keough M.J. (2002) Experimental design and data analysis for biologists, pp. 92-98, 111-130
- ▶ Zuur, A.F. et al. 2009. Mixed effects models and extensions in ecology with R. - Statistics for biology and health. Springer, New York, NY.