Язык R
лекция 4

Артем Артемов, Елена Ставровская, Светлана Виноградова

4 октября 2014

Династия

Фонд Дмитрия Зимина

ИЛС

ИнтерЛабСервис

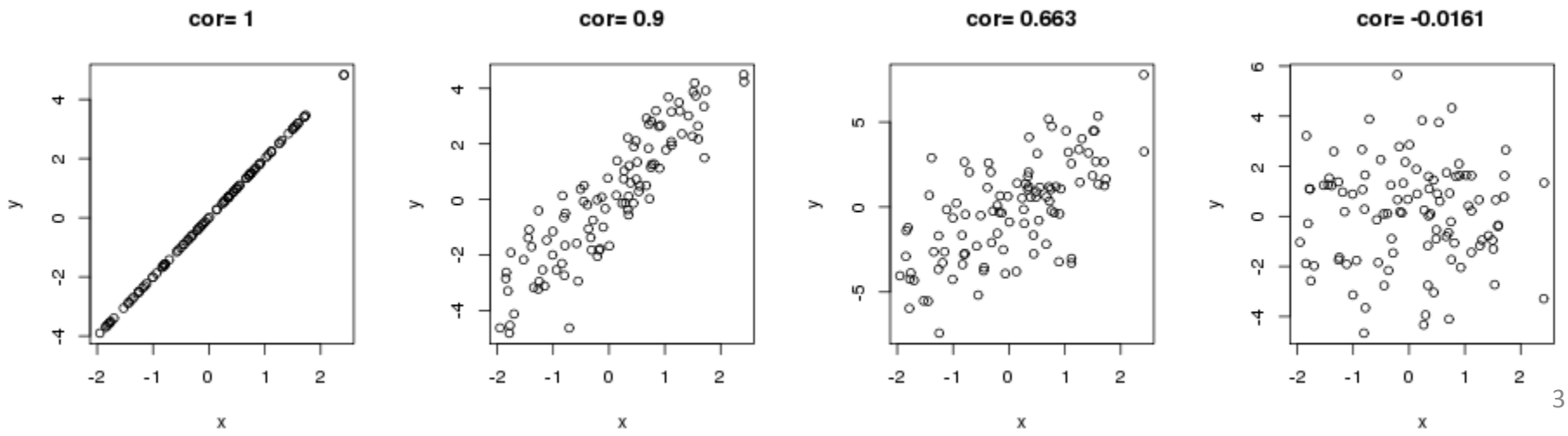
Outline

- `lm(y ~ x)`
- Модели с множеством предикторов
  - `lm(y ~ x1 + x2 + x3)`
  - Если *x* – не число, а фактор?
  - Взаимодействия факторов
  - ANOVA
- Предсказания по модели и валидация моделей:
  - Предсказание *y* по $x_1 \ldots x_n$, зная модель
  - Сравнение и валидация моделей, переобученность
- Обобщенные линейные модели (glm)

Корреляция

- Измеряет линейную зависимость между переменными
- Не означает причинно-следственной связи

```
> cor(x, y)
> cor(x, y, method="spearman")   #ранговая корреляция
> cor(m)
```



Dataset

- Цены на ноутбуки (реальный прайс-лист)
- Что определяет цену, как она зависит от разных параметров?
- Как предсказать цену, зная параметры ноутбука?

```
> laptop=read.csv("laptop_price.csv")
> head(laptop)
```

	Manufacturer	Model	Processor	Memory_Gb	HDD_Gb	HDD_type	Price_RUR
1	Acer	Aspire	i3-3110M	4	500	HDD	16400
2	Acer	Aspire	i3-3120M	4	500	HDD	16500
3	Acer	Aspire	i5-3230M	4	500	HDD	18500
4	Acer	Aspire	C-70	2	500	HDD	12000
5	Acer	Aspire	C-70	2	500	HDD	12000
6	Acer	Aspire	1007U	2	500	HDD	11300
7	Acer	Aspire	i5-2467M	4	240	SSD	33800

	Screen_size_inch	Battery_capacity_mAh	OS	Color
1	15.6	4400	Win8	black
2	15.6	4400	Win8	black
3	15.6	4400	Win8	black
4	11.6	2500	Win8	turquoise
5	11.6	2500	Win8	black
6	11.6	5000	Win8	turquoise
7	13.3	3260	Win7HP	silver

Корреляция

Если на входе – матрица, cor вычисляет корреляции между всеми колонками матрицы

```
> m=as.matrix(laptop[,c("Memory_Gb", "HDD_Gb",
  "Screen_size_inch", "Battery_capacity_mAh")])
> cor(m)
```

	Memory_Gb	HDD_Gb	Screen_size_inch	Battery_capacity_mAh
Memory_Gb	1.000	0.6741	0.5259	0.2282
HDD_Gb	0.674	1.0000	0.5156	0.0568
Screen_size_inch	0.526	0.5156	1.0000	-0.0329
Battery_capacity_mAh	0.228	0.0568	-0.0329	1.0000

Регрессия

Основная идея: наблюдаемые значения зависимой переменной – измерения, которые содержат шум

$$y = f(x, b) + e$$

b_i – параметры модели

x_i – предикторы (независимые переменные)

e – ошибка (все, что мы не можем измерить и учесть в модели)

Мат. ожидание **E**[e] = 0

Линейная регрессия

$$y = f(x, b) + e$$

$$f(x, b) = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_k x_k$$

b_i – параметры модели

x_i – предикторы (независимые переменные)

e – ошибка (все, что мы не можем измерить и учесть в модели)

e распределено нормально!

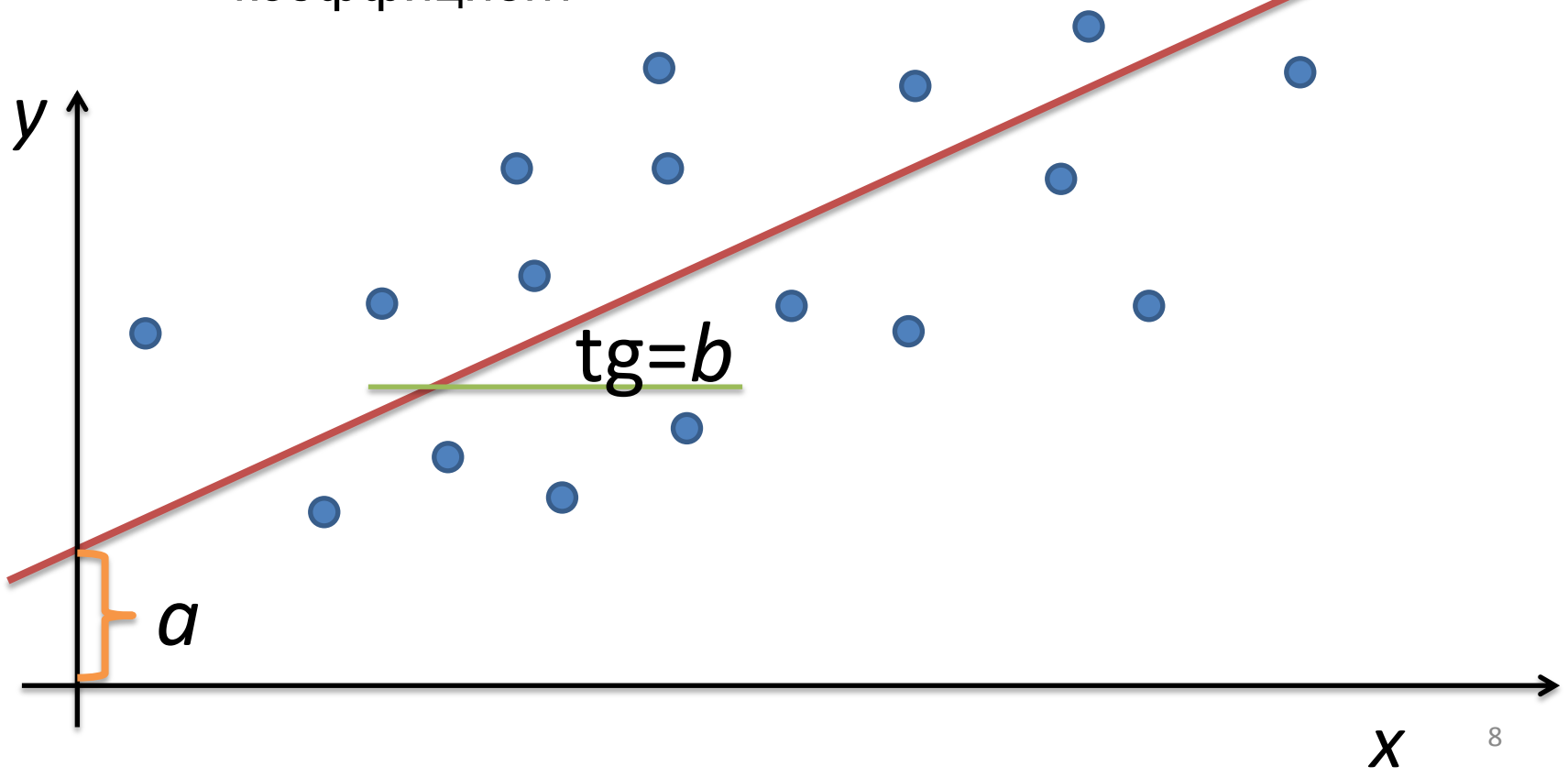
Линейная регрессия с 1 переменной

В случае одной независимой переменной

– константа

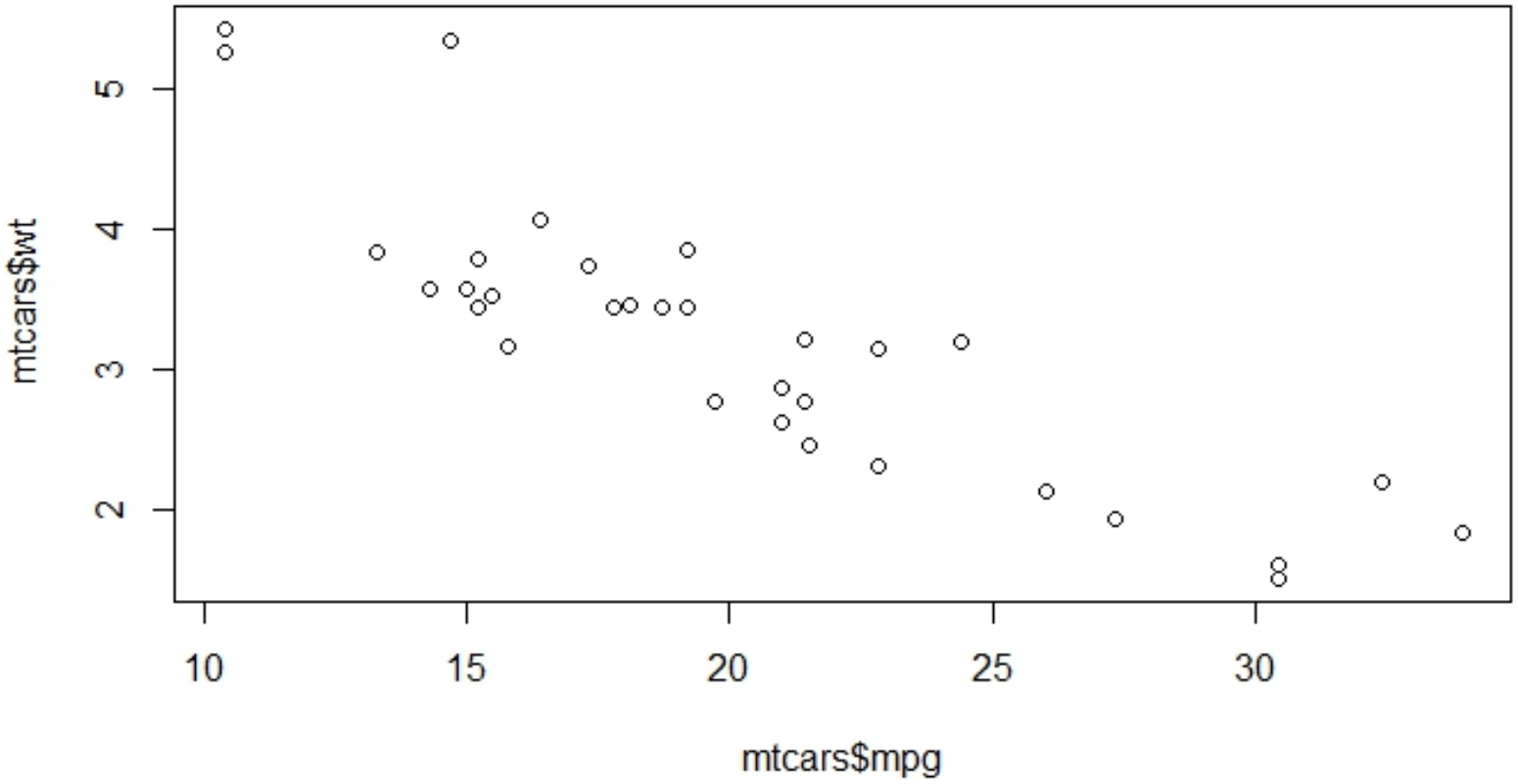
– коэффициент

$$f(x, b) = a + b\,x$$



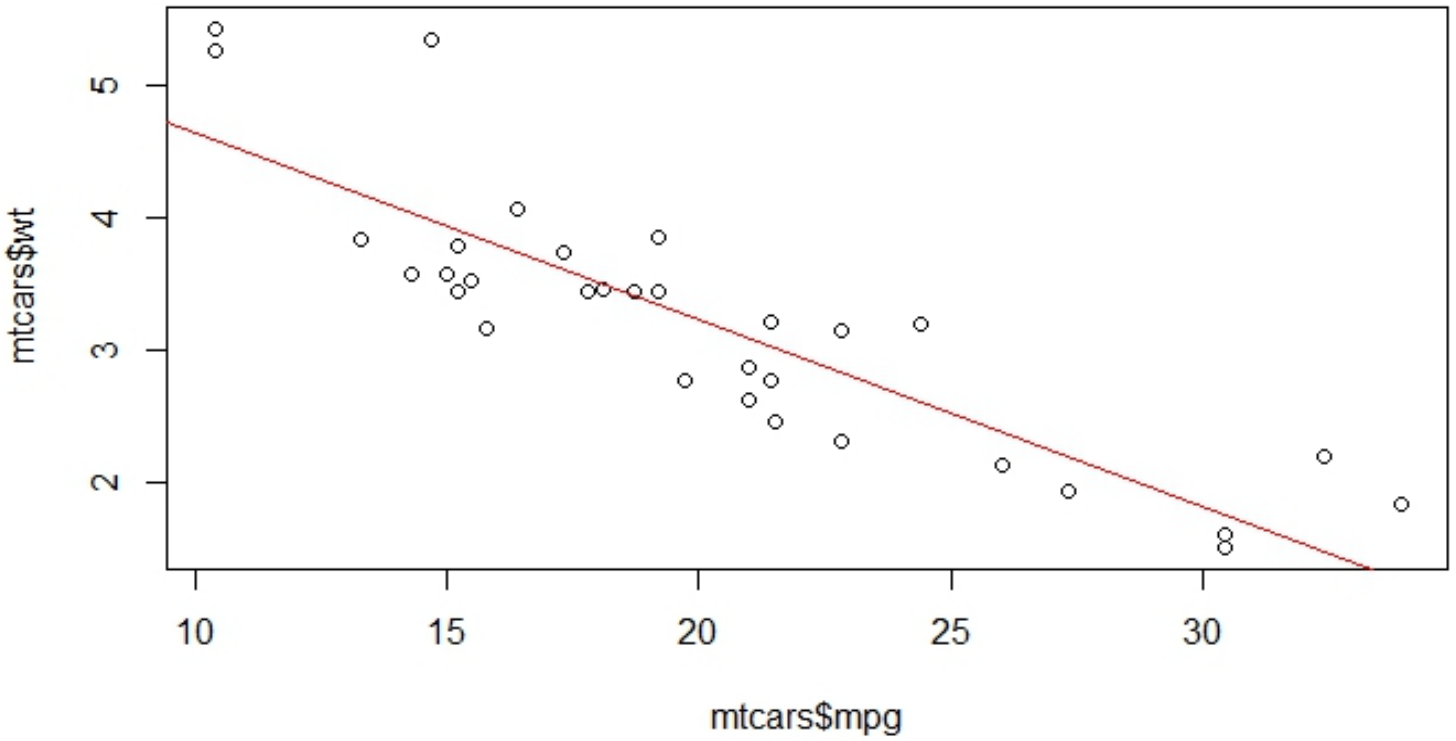
Модель с 1 предиктором. Пример 1

```
> plot(mtcars$mpg, mtcars$wt)
```



Модель с 1 предиктором. Пример 1

```
> plot(mtcars$mpg, mtcars$wt)
> lm1<-lm(mtcars$wt~mtcars$mpg)
> abline(lm1, col='red')
```



Модель с 1 предиктором. Пример 1

```
> lm1<-lm(mtcars$wt~mtcars$mpg)
> lm1

Call:
lm(formula = mtcars$wt ~ mtcars$mpg)

Coefficients:
(Intercept)   mtcars$mpg
     6.0473      -0.1409
```

Модель с 1 предиктором. Пример 1

Коэффициенты:

```
> lm1$coefficients
(Intercept)  mtcars$mpg
   6.047255   -0.140862
> lm1$coefficients[1]
(Intercept)
   6.047255
```

Оценка качества линейной регрессионной модели

```
> summary(lm1)
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.04726    0.30869  19.590  < 2e-16 ***
mtcars$mpg   -0.14086    0.01474  -9.559 1.29e-10 ***
---
Signif. Codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’
  0.1 ‘ ’ 1
```

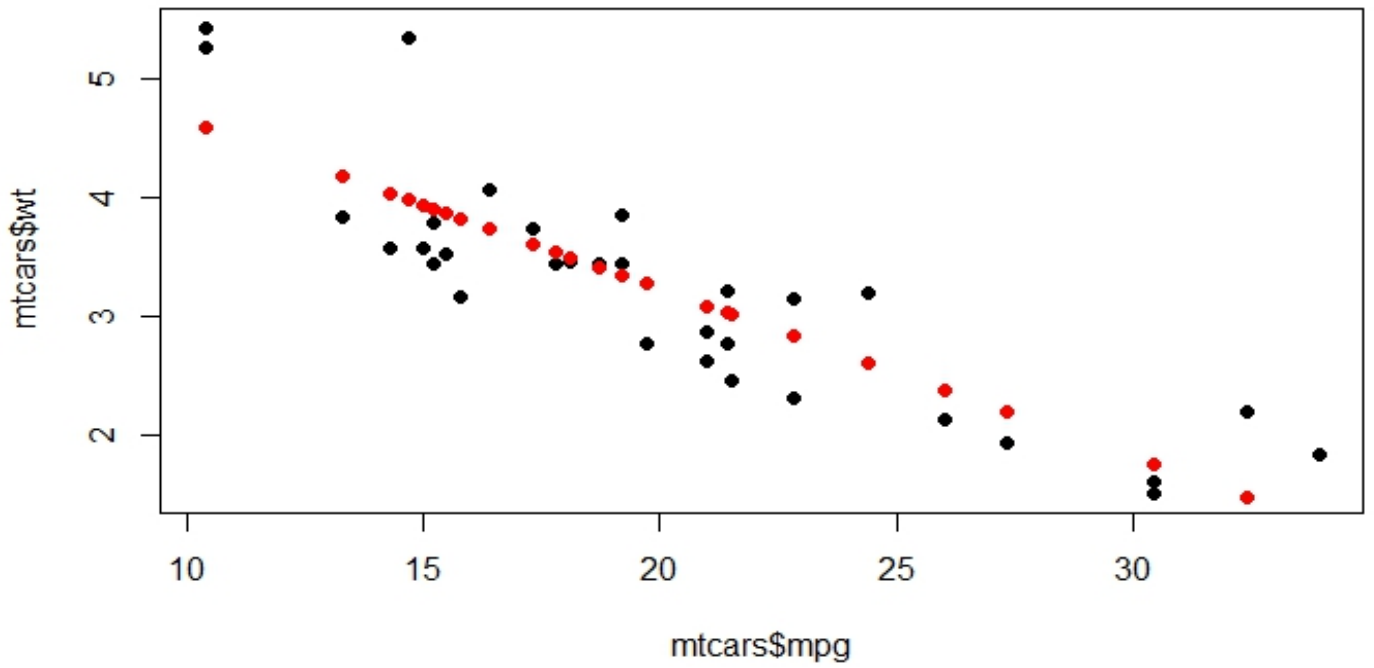
оценка параметра

t-статистика (оценка/стандарная ошибка)

p-value

Предсказания модели = fitted values

```
> plot(mtcars$mpg, mtcars$wt, pch=19)
> lm1<-lm(mtcars$wt~mtcars$mpg)
> points(mtcars$mpg, lm1$fitted, pch=19,
   col='red')
```



Немного теории

- Есть вариация (=дисперсия) в *y*, которую пытаемся объяснить дисперсией в *x*. *SST (=SStotal)*
- По *x* можно предсказать y_{pred}. Если x – фактор, то y_{pred} – просто среднее по группе.
- Вариация y_{pred} – вариация *y*, объясненная иксом. SSX (=*SSexplained_by_X)*
- SST=SSX+SSE

$$y = f(x, b) + e$$

$$F = \frac{MS_{\text{Treatments}}}{MS_{\text{Error}}} = \frac{SS_{\text{Treatments}}/(I-1)}{SS_{\text{Error}}/(n_T - I)}$$

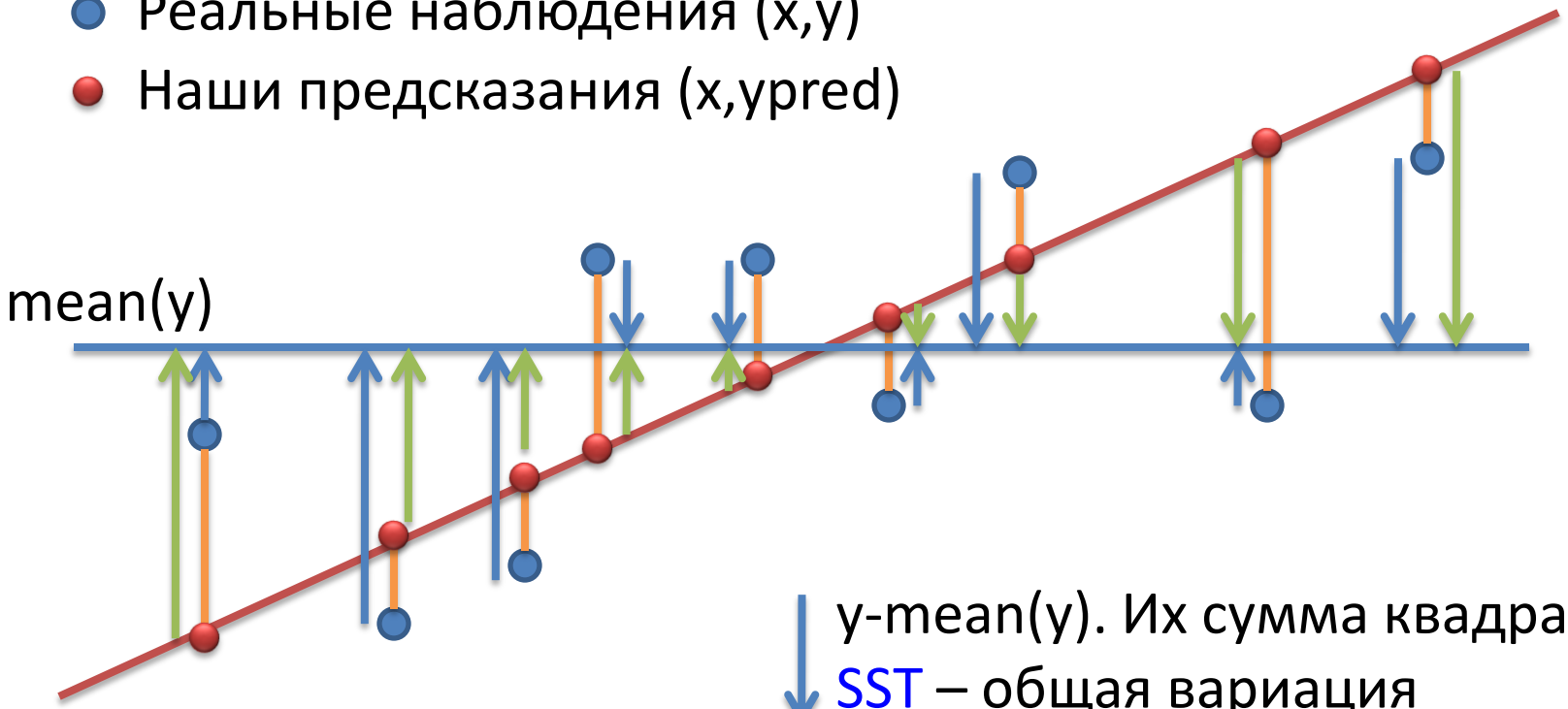
- R^2=SSX/SST

I уровней фактора

n_T всего точек

SST = SSX + SSE

- Реальные наблюдения (x,y)
- Наши предсказания (x,ypred)



y-mean(y). Их сумма квадратов SST – общая вариация

y-ypred. Ошибки. SSE – необъясненная вариация

ypred-mean(y). Ошибки. SSX – объясненная вариация

- R^2=SSX/SST
- F=(SSX)/
- /(SSE/(npoints-nlevels))

Оценка качества линейной регрессионной модели

Доля объясненной дисперсии (чем ближе к 1, тем лучше)

```
> summary(lm1)
...
```

$$R^2 = \frac{Var(\hat{Y})}{Var(Y)}$$ =SSX/SST

```
Multiple R-squared: 0.7528,    Adjusted R-
  squared: 0.7446
F-statistic: 91.38 on 1 and 30 DF,  p-value:
  1.294e-10
```

F-статистика (отношение объясненной дисперсии к ошибочной)

P-value для всей модели

$$F = \frac{Var(\hat{Y})}{Var(error)}$$ =(SSX)/ (SSE/(npoints-nlevels))

Построение модели с несколькими предикторами

Шаг 1. Модель с 1 предиктором

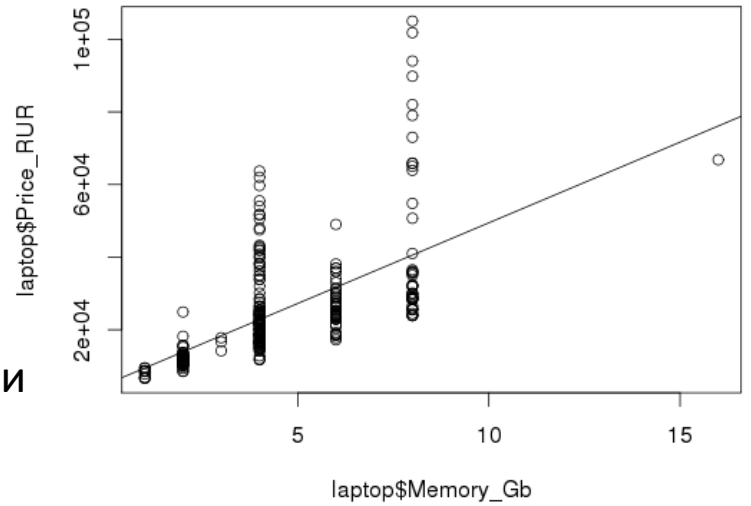
```
> L_M=lm(Price_RUR ~ Memory_Gb, data=laptop)
> summary(L_M)
…
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     5023       1651    3.04   0.0026 **
Memory_Gb       4442        333   13.34   <2e-16 ***
…
Residual standard error: 12300 on 304 degrees of
  freedom
Multiple R-squared: 0.369,    Adjusted R-squared: 0.367

F-statistic:  178 on 1 and 304 DF,  p-value: <2e-16
```

Корреляция и регрессия

```
> cor.test(laptop$Price_RUR, laptop$Memory_Gb)

   Pearson's product-moment correlation

data:  laptop$Price_RUR and laptop$Memory_Gb
t = 13.3, df = 304, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.532 0.674
sample estimates:
  cor
0.608
```

```
> 0.608^2
[1] 0.37
```

R2 равен квадрату коэффициента корреляции



Несколько предикторов

- Как цена ноутбука зависит от объема памяти, объема жесткого диска и размера дисплея?
- Предикторы разделены через **+**

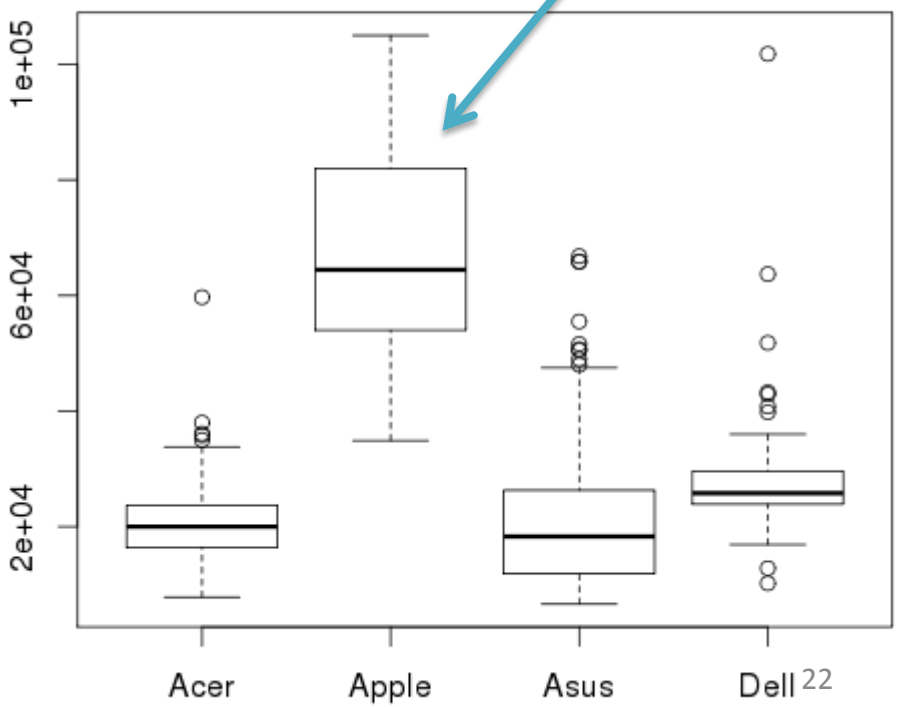
```
> l_MSH=lm(Price_RUR ~ Memory_Gb + Screen_size_inch +
  HDD_Gb, data=laptop)
> summary(l_MSH)
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      16475.481   4437.238   3.713 0.000244 ***
Memory_Gb         7266.167    406.667  17.868  < 2e-16 ***
Screen_size_inch  -511.390    350.186  -1.460 0.145237
HDD_Gb             -31.022      3.305  -9.387  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10650 on 302 degrees of freedom
Multiple R-squared: 0.5308,   Adjusted R-squared: 0.5262
F-statistic: 113.9 on 3 and 302 DF,  p-value: < 2.2e-16
```

Если x – фактор? Пример

- Уже умеем искать значимые отличия при разных уровней факторов

```
> boxplot(Price_RUR ~ Manufacturer,
  data=laptop)
```

Guess who



Верно ли, что для хотя бы одного уровня фактора наблюдаем отличия?

Если x – фактор? Модель

- Почему одна переменная превратилась в несколько?

```
> L_M=lm(Price_RUR ~ Manufacturer, data=laptop)
> summary(L_M)
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        21198.6     1415.0  14.981  < 2e-16 ***
ManufacturerApple  46078.5     3527.5  13.063  < 2e-16 ***
ManufacturerAsus     206.2     1714.4   0.120 0.904357
ManufacturerDell    7427.6     2079.1   3.573 0.000411 ***
---
Residual standard error: 12090 on 302 degrees of freedom
Multiple R-squared: 0.3958,      Adjusted R-squared:
0.3898
F-statistic: 65.95 on 3 and 302 DF,  p-value: < 2.2e-16
```

Если x – фактор?

- Влияет ли цвет ноутбука на его цену?
- Модель, если x – число: $y_i = \alpha x_{1i} + \beta x_{2i} + \varepsilon_i$
- Если x – фактор, то такая запись не подходит. Вместо этого:

$$y_i = \alpha_1 I(x_{1i}==black) + \alpha_2 I(x_{1i}==white) + \ldots + \varepsilon_i$$

Коэффициент (подбираются при построении модели)

Индикатор (равен 1, если x – черный цвет, иначе 0)

Если две факторные переменные?

$$y_i = \alpha_1 I(x_{1i}==black) + \alpha_2 I(x_{1i}==white) + \ldots +$$

$$+ \beta_1 I(x_{2i}==Apple) + \beta_2 I(x_{2i}==ASUS) + \ldots + \varepsilon_i$$

Числа + факторы

- Шаг 1. Как зависит цена ноутбука от размера жесткого диска?

#Постоим lm с 1 переменной

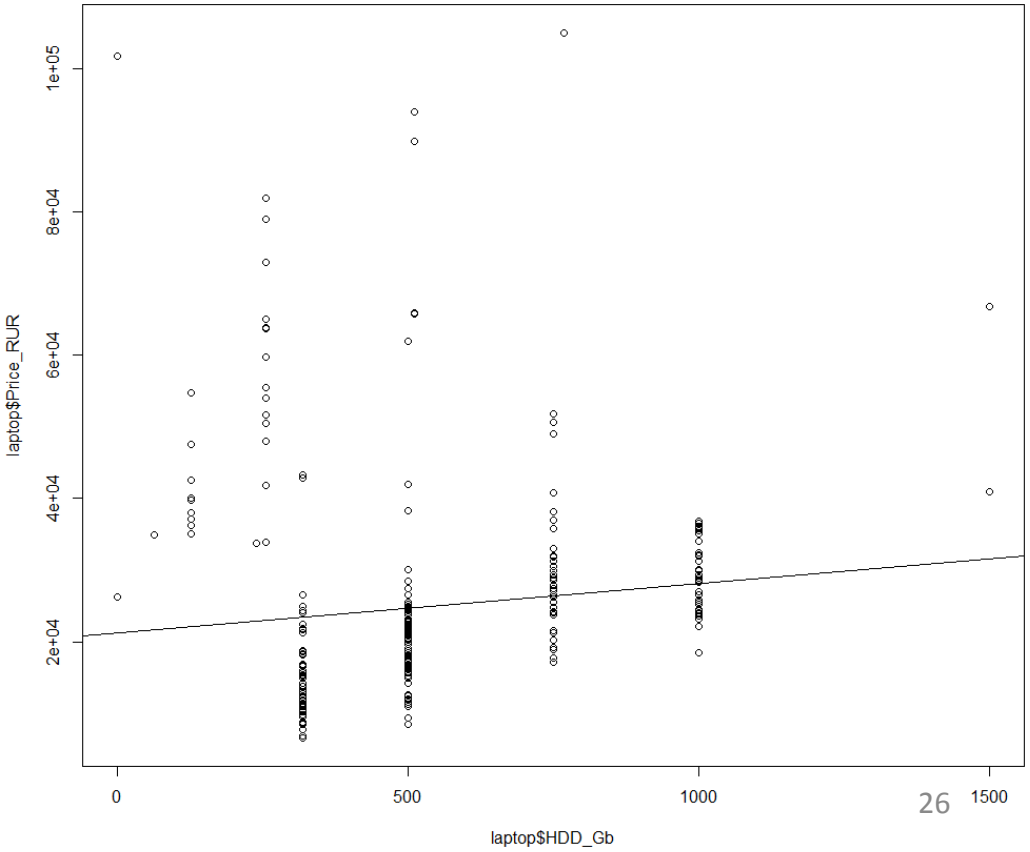
```
> l1=lm(Price_RUR ~ HDD_Gb, data=laptop)
> summary(l1)
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 21238.584   2027.553  10.475   <2e-16 ***
HDD_Gb          6.913      3.410   2.027   0.0435 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
…
```

#0.043 – на грани порога значимости

Числа + факторы

#Нарисуем scatterplot

```
> plot(laptop$HDD_Gb, laptop$Price_RUR)
> abline(l1)
```



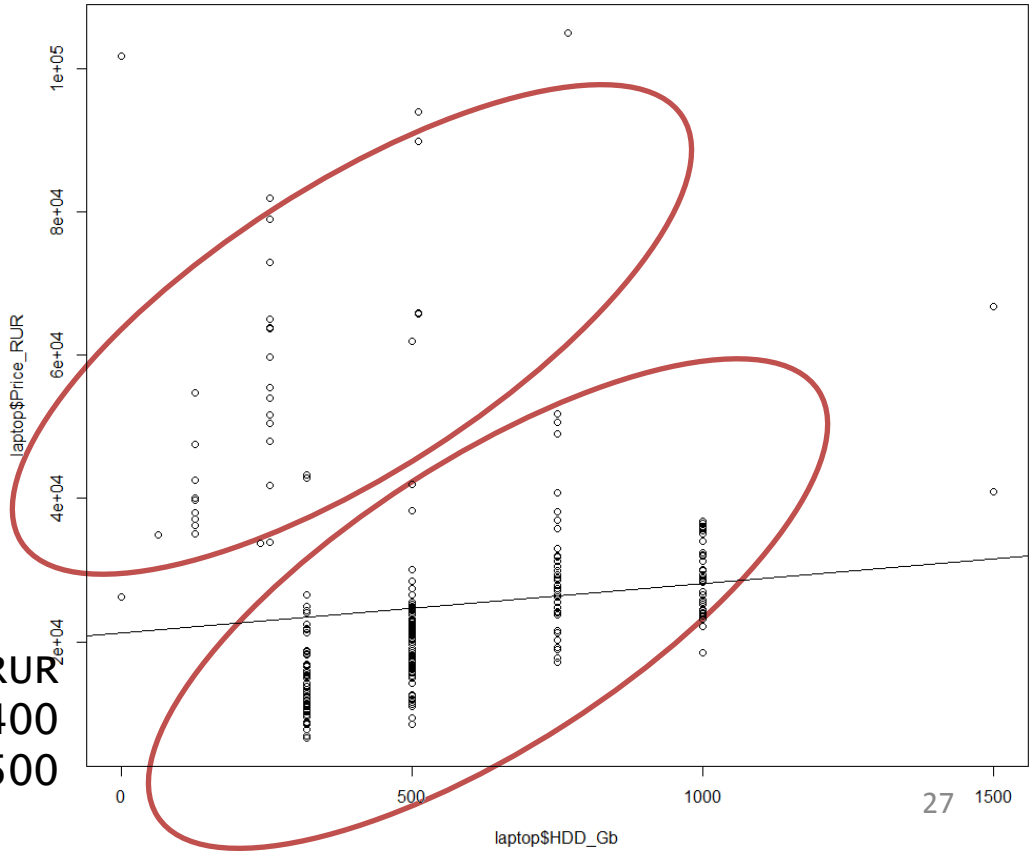
Числа + факторы

#Нарисуем scatterplot

```
> plot(laptop$HDD_Gb, laptop$Price_RUR)
> abline(l1)
```

Видно 2 группы

Это знак, что мы чего-то не учли



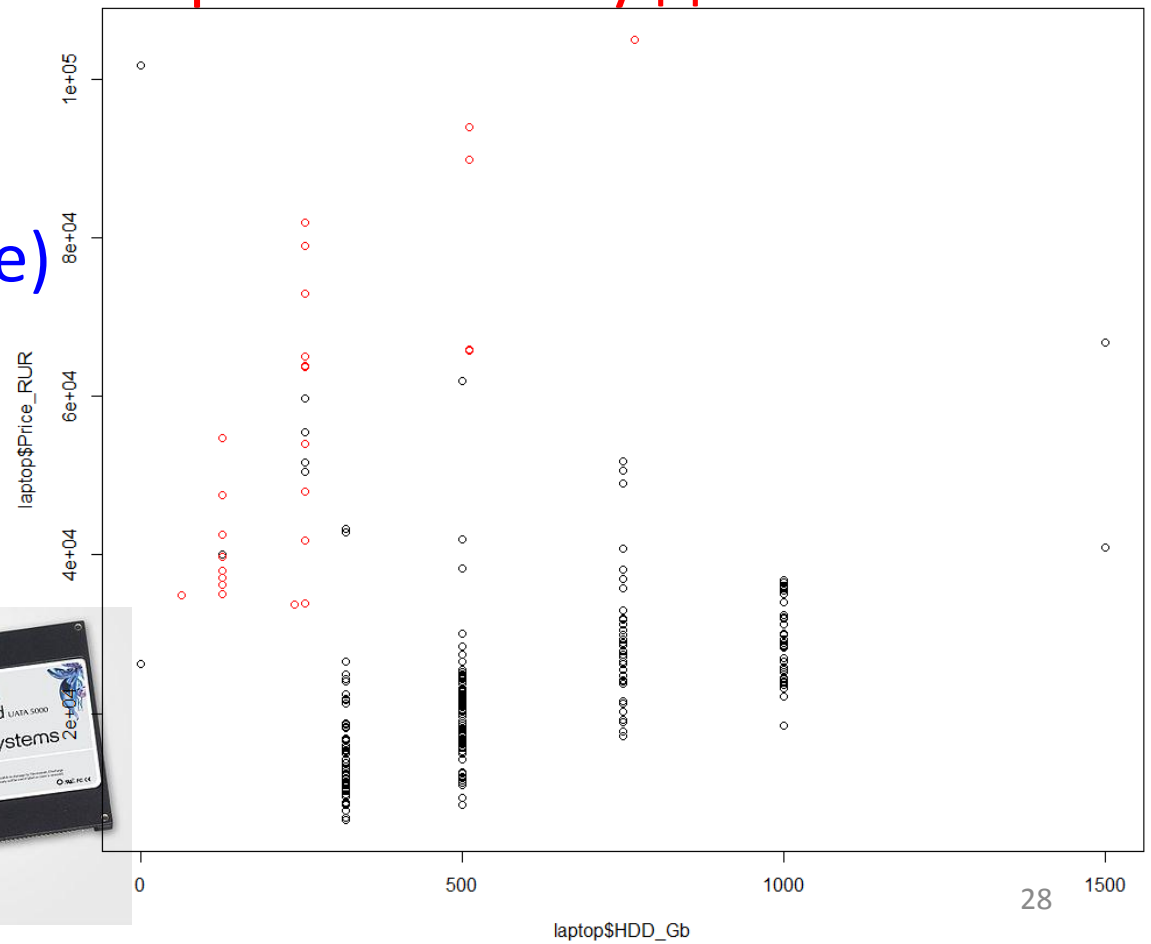
Memory_Gb	HDD_Gb	HDD_type	Price_RUR
4	500	HDD	16400
4	500	HDD	18500

Числа + факторы

Шаг 2. Фактор, который мы не учли – тип накопителя, HDD или SSD. Вторые сильно дороже

```
# Нарисуем scatterplot и покрасим по типу диска
> plot(laptop$HDD_Gb,
    laptop$Price_RUR,
    col=laptop$HDD_type)
```

Похоже, мы правы



Числа + факторы

```
#Добавим тип диска как переменную в модель
> l2=lm(Price_RUR ~ HDD_Gb + HDD_type, data=laptop)
> summary(l2)
…
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 10741.160   1594.347   6.737 8.14e-11 ***
HDD_Gb         20.290      2.591   7.830 8.27e-14 ***
HDD_typeSSD 40797.575   2442.199  16.705  < 2e-16 ***
…
```

Значимость улучшилась

Наклон прямой будет одинаковым, но среднее между группами - отличается

Числа + факторы

#Нарисуем scatterplot и две регрессионные прямые (для каждого значения фактора)

```
> l2$coeff
(Intercept)      HDD_Gb HDD_typeSSD
10741.15975    20.28962 40797.57542
```

```
> plot(laptop$HDD_Gb,
   laptop$Price_RUR,
   col=laptop$HDD_type)
```

```
> abline(l2$coeff[1], l2$coeff[2],
   col="black")
```

```
> abline(l2$coeff[1]+l2$coeff[3],
   l2$coeff[2], col="red")
```

Price=10741+20**HDD_Gb*+40797*I(*type*=SSD)

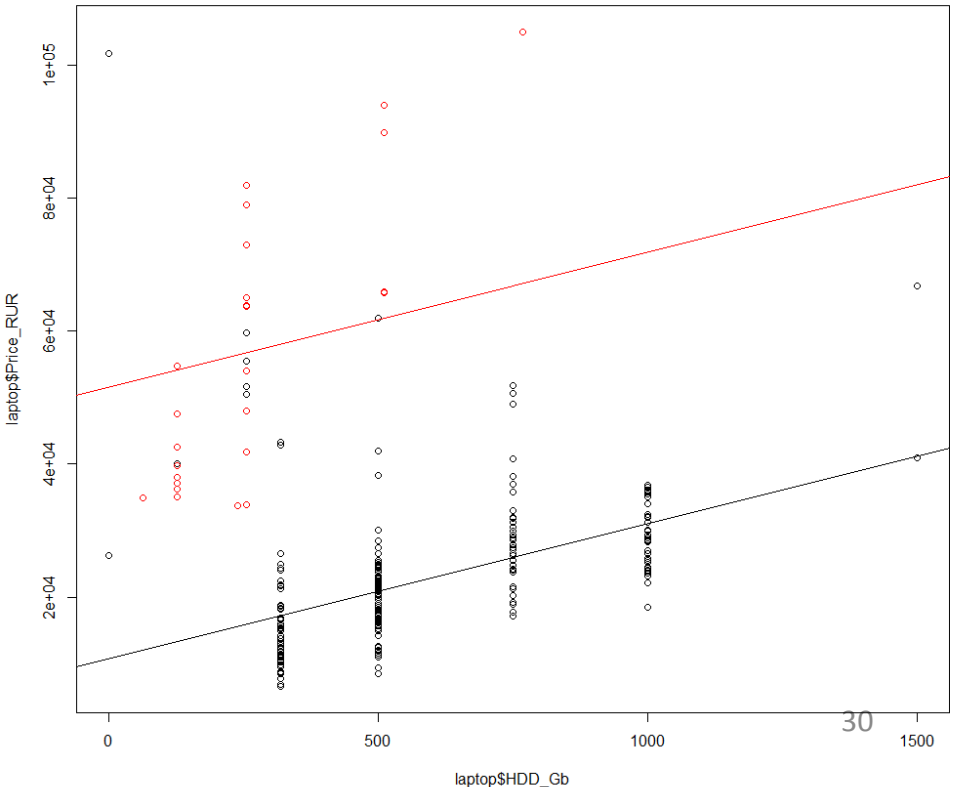
if *type*≠SSD:

Price=10741+20**HDD_Gb*+~~40797*0~~

if *type*=SSD:

Price=10741+20**HDD_Gb*+40797*1

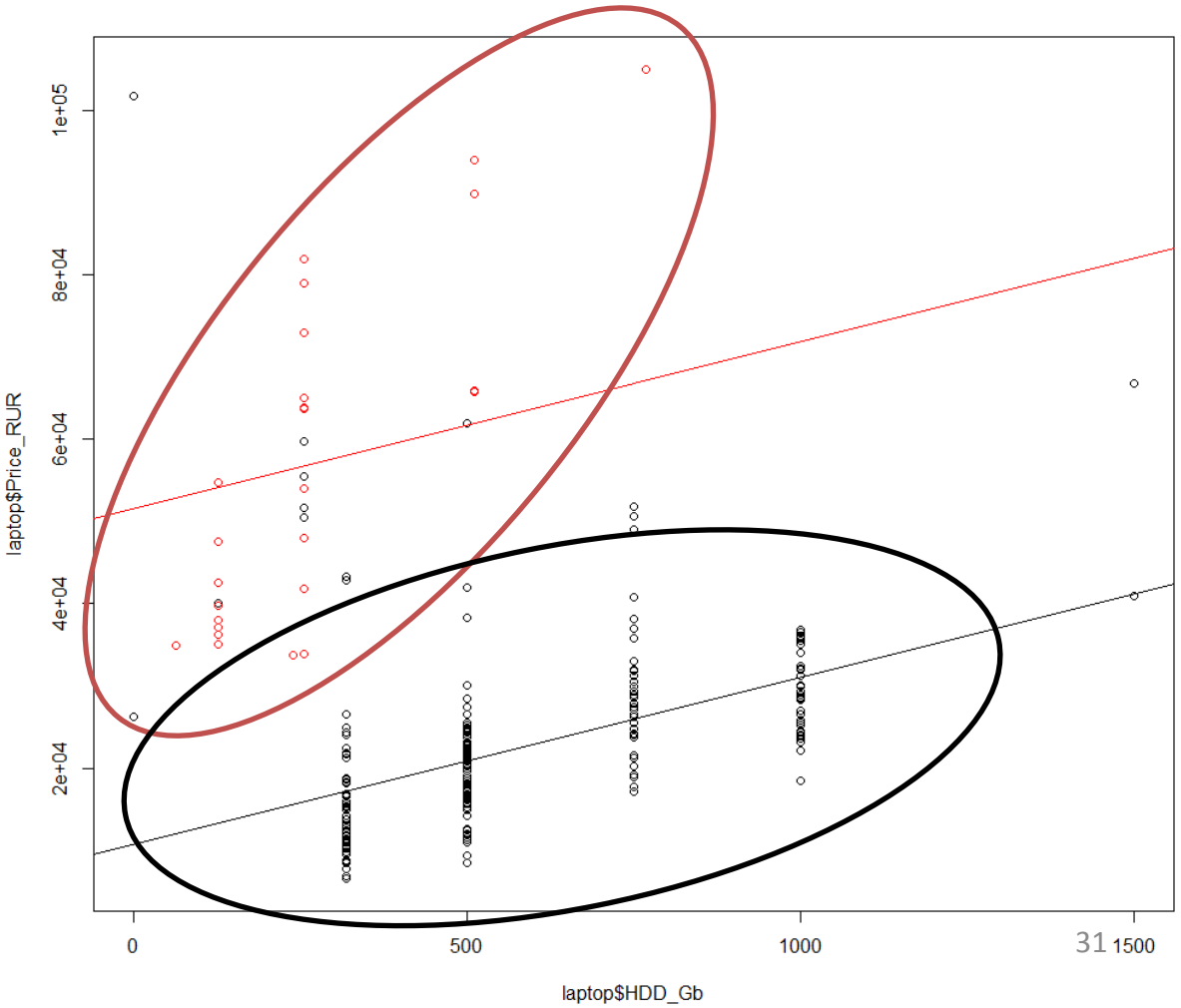
=(10741+40797)+20**HDD_Gb*



Числа + факторы. Взаимодействие переменных

Шаг 3.

- Похоже, что наклоны для двух групп тоже отличаются, другими словами, каждый 1Gb SSD стоит дороже каждого 1Gb HDD.
- Как это учесть?



Числа + факторы. Взаимодействие переменных

- Как в формуле сделать разные наклоны для разных групп факторов?
- Было: *Price*=a+b**HDD_Gb*+c*I(*type*=SSD)
- Надо:

Price=a+

+(b1*I(*type*=SSD)+b2*I(*type*=HDD))**HDD_Gb*+

+c*I(*type*=SSD)

=a+

+(b1*I(*type*=SSD)+b2(1-I(*type*=SSD)))**HDD_Gb*+

+c*I(*type*=SSD)

…преобразуем формулу…

Числа + факторы. Взаимодействие переменных

```
> l3=lm(Price_RUR ~ HDD_Gb + HDD_type + HDD_Gb:HDD_type, data=laptop)
> l3=lm(Price_RUR ~ HDD_Gb*HDD_type , data=laptop)
> summary(l3)
```

Эквивалентные записи:
a*b := a + b + a:b

```
Call:
lm(formula = Price_RUR ~ HDD_Gb * HDD_type, data = laptop)

Residuals:
   Min     1Q Median     3Q    Max
-21886  -6049  -1461   2885  89344

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        12430.529   1525.776   8.147 9.97e-15 ***
HDD_Gb                17.270      2.488   6.941 2.38e-11 ***
HDD_typeSSD        18232.081   4265.934   4.274 2.58e-05 ***
HDD_Gb:HDD_typeSSD    80.870     12.874   6.281 1.17e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10480 on 302 degrees of freedom
Multiple R-squared: 0.5457,        Adjusted R-squared: 0.5412
F-statistic: 120.9 on 3 and 302 DF,  p-value: < 2.2e-16
```

Взаимодействие значимо

Числа + факторы. Взаимодействие переменных

```
> l3$coeff
      (Intercept)            HDD_Gb      HDD_typeSSD HDD_Gb:HDD_typeSSD
      12430.52872         17.26953      18232.08144           80.86960
```

Нарисуем регрессионные прямые для каждого из значений факторной переменной

Price=12430+17**HDD_Gb*+18232*I(*type*=SSD)+81**HDD_Gb**I(*type*=SSD)

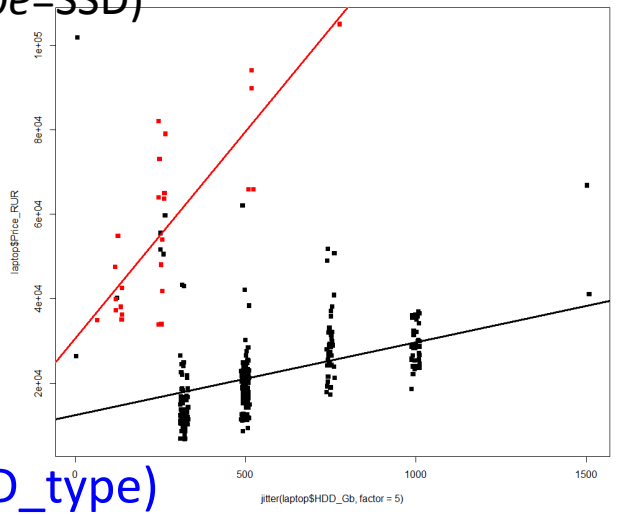
if *type*≠SSD:

Price=12430+17**HDD_Gb*+~~18232*0+81**HDD_Gb**0~~

if *type*=SSD:

Price=12430+17**HDD_Gb*+18232*1+81**HDD_Gb**1=

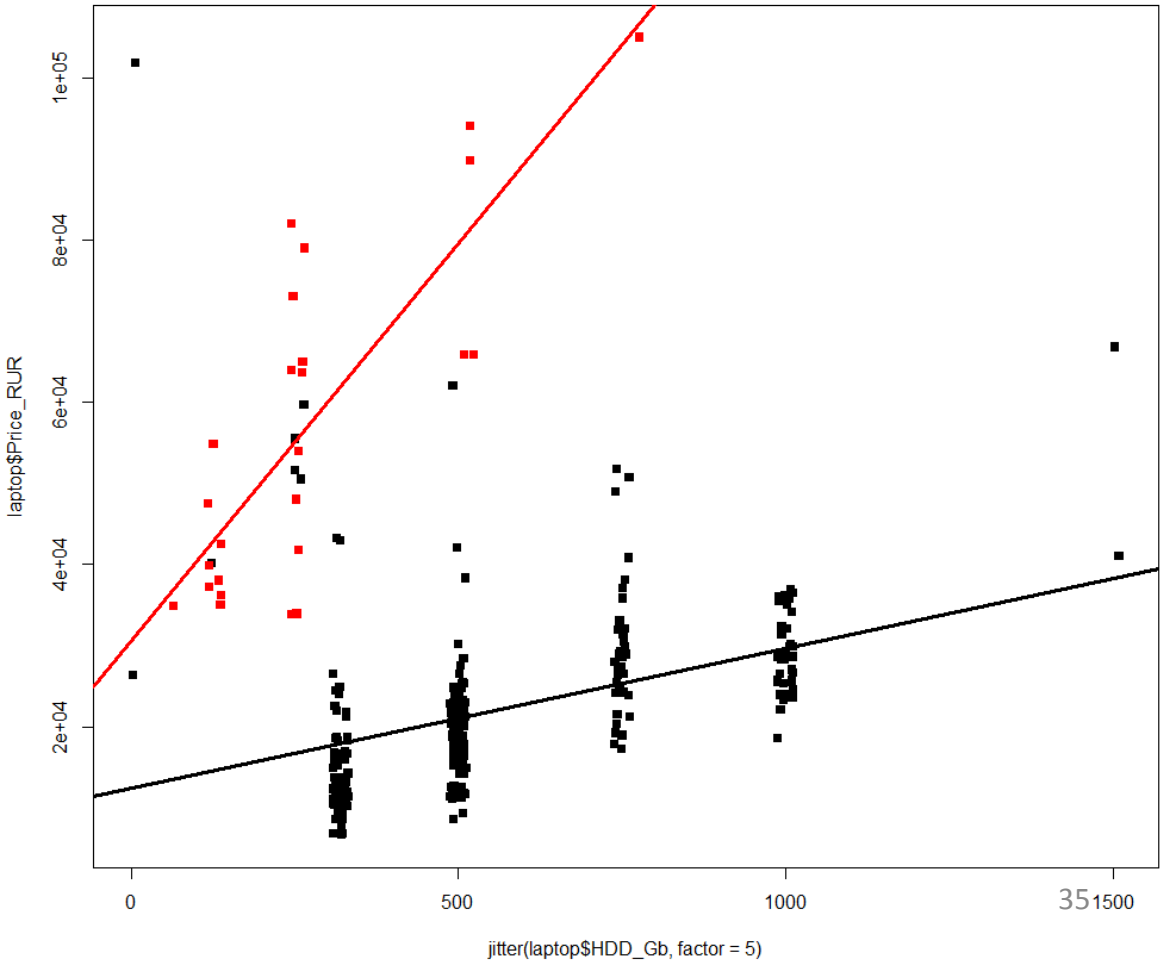
=(12430+18232)+ (17+81)**HDD_Gb*




```
> plot(laptop$HDD_Gb, laptop$Price_RUR, col=laptop$HDD_type)
> abline(l3$coeff[1], l3$coeff[2])
> abline(l3$coeff[1]+l3$coeff[3], l3$coeff[2]+l3$coeff[4], col="red")
```

Числа + факторы.
Взаимодействие переменных

```
> plot(laptop$HDD_Gb, laptop$Price_RUR, col=laptop$HDD_type)
> abline(l3$coeff[1], l3$coeff[2])
> abline(l3$coeff[1]+l3$coeff[3], l3$coeff[2]+l3$coeff[4], col="red")
```



Обозначения в формулах

a*b = a + b + a:b

- y ~ x + 0
- y ~ x – 1

– x1 удаляет предиктор x1 из модели

- y ~ a*b – a
- y ~ b + a:b

- y ~ . # . – все остальные переменные

- I(a+b), I(a*b) #защитить арифметические операторы

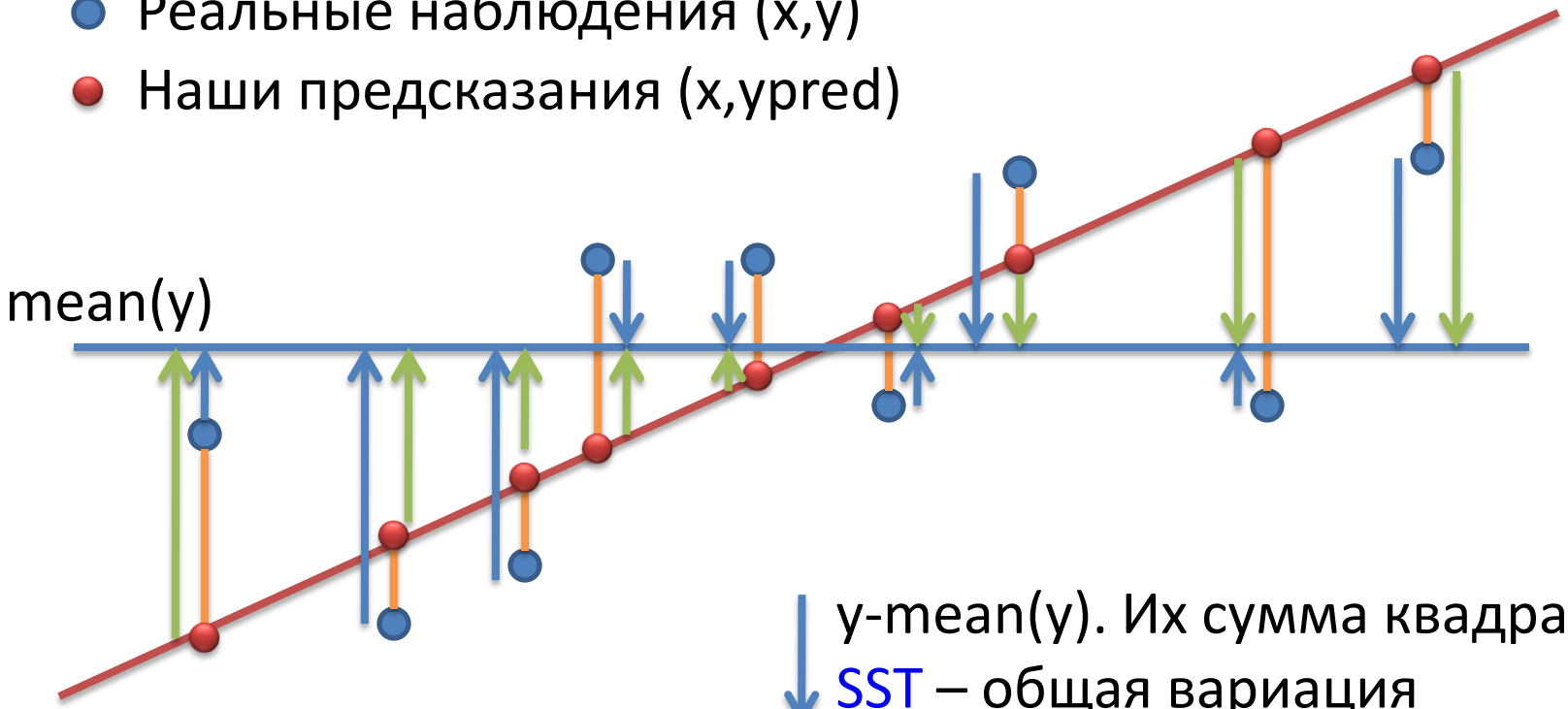
Какая из моделей лучше?

- Можно придумать разные модели, одна учитывает объем памяти, другая – ещё объем жесткого диска, третья дополнительно учитывает, является ли диск диском или твердотельным накопителем (SSD).
- Как сравнить, какая лучше?
- Наивный подход: насколько хорошо модель описывает данные ≈ насколько мала необъясненная дисперсия в y ≈ насколько R^2 близок к 1. Не работает, т.к. добавление параметров увеличивает R^2
- Скорректированный R^2 (*adjusted R^2*), информационные критерии (*AIC*, *BIC*)

http://ru.wikipedia.org/wiki/Коэффициент_детерминации

SST = SSX + SSE (повторение)

- Реальные наблюдения (x,y)
- Наши предсказания (x,ypred)



y-mean(y). Их сумма квадратов
SST – общая вариация

y-ypred. Ошибки.
SSE – необъясненная вариация

ypred-mean(y). Ошибки.
SSX – объясненная вариация

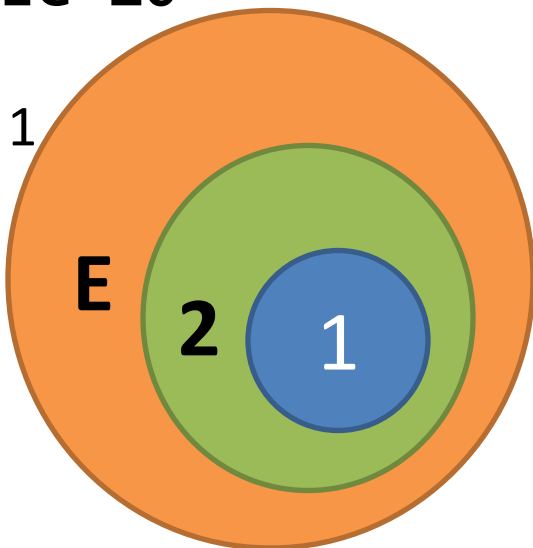
- R^2=SSX/SST
- F=((SSX)/(nparam-1))/
- /(SSE/(npoints-nparam))

ANOVA для сравнения моделей

```
> fit2=lm(Price_RUR ~ Memory_Gb+HDD_Gb+HDD_type, data=laptop)
> fit1=lm(Price_RUR ~ Memory_Gb, data=laptop)
> anova(fit1, fit2)
Analysis of Variance Table
Model 1: Price_RUR ~ Memory_Gb + HDD_Gb + HDD_type
Model 2: Price_RUR ~ Memory_Gb
  Res.Df      RSS Df Sum of Sq   F Pr(>F)
1    302 2.56e+10
2    304 4.61e+10 -2 -2.05e+10 121 <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

$$F = \frac{\left(\frac{\mathrm{RSS}_1 - \mathrm{RSS}_2}{p_2 - p_1}\right)}{\left(\frac{\mathrm{RSS}_2}{n - p_2}\right)}$$

- F=((SSX**2**-SSX**1**)/(nparam**2**-nparam**1**))/ /(SSE2/(npoints-nparam**2**))



http://www.statmethods.net/stats/regression.html
http://en.wikipedia.org/wiki/F_test

ANOVA для сравнения моделей

- Противоположный пример

```
> fit1=lm(Price_RUR ~ Memory_Gb+HDD_Gb+HDD_type+Color,
    data=laptop)
> fit2=lm(Price_RUR ~ Memory_Gb+HDD_Gb+HDD_type, data=laptop)
> anova(fit1, fit2)
Analysis of Variance Table

Model 1: Price_RUR ~ Memory_Gb + HDD_Gb + HDD_type + Color
Model 2: Price_RUR ~ Memory_Gb + HDD_Gb + HDD_type
Res.Df      RSS  Df Sum of Sq    F Pr(>F)
1    288 2.43e+10
2    302 2.56e+10 -14 -1.27e+09 1.08   0.38
```

Немного теории

- Есть вариация (=дисперсия) в *y*, которую пытаемся объяснить дисперсией в *x*. *SST (=SStotal)*
- По *x* можно предсказать y_{pred}. Если x – фактор, то y_{pred} – просто среднее по группе.
- Вариация y_{pred} – вариация *y*, объясненная иксом. SSX (=*SSexplained_by_X)*
- SST=SSX+SSE

$$F = \frac{MS_{\text{Treatments}}}{MS_{\text{Error}}} = \frac{SS_{\text{Treatments}}/(I-1)}{SS_{\text{Error}}/(n_T - I)}$$

- R^2=SSX/SST

I уровней фактора

n_T всего точек

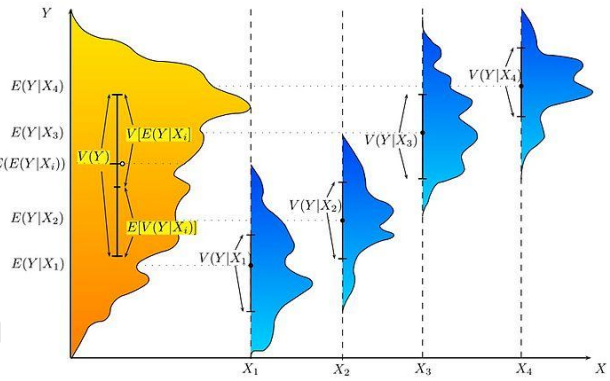
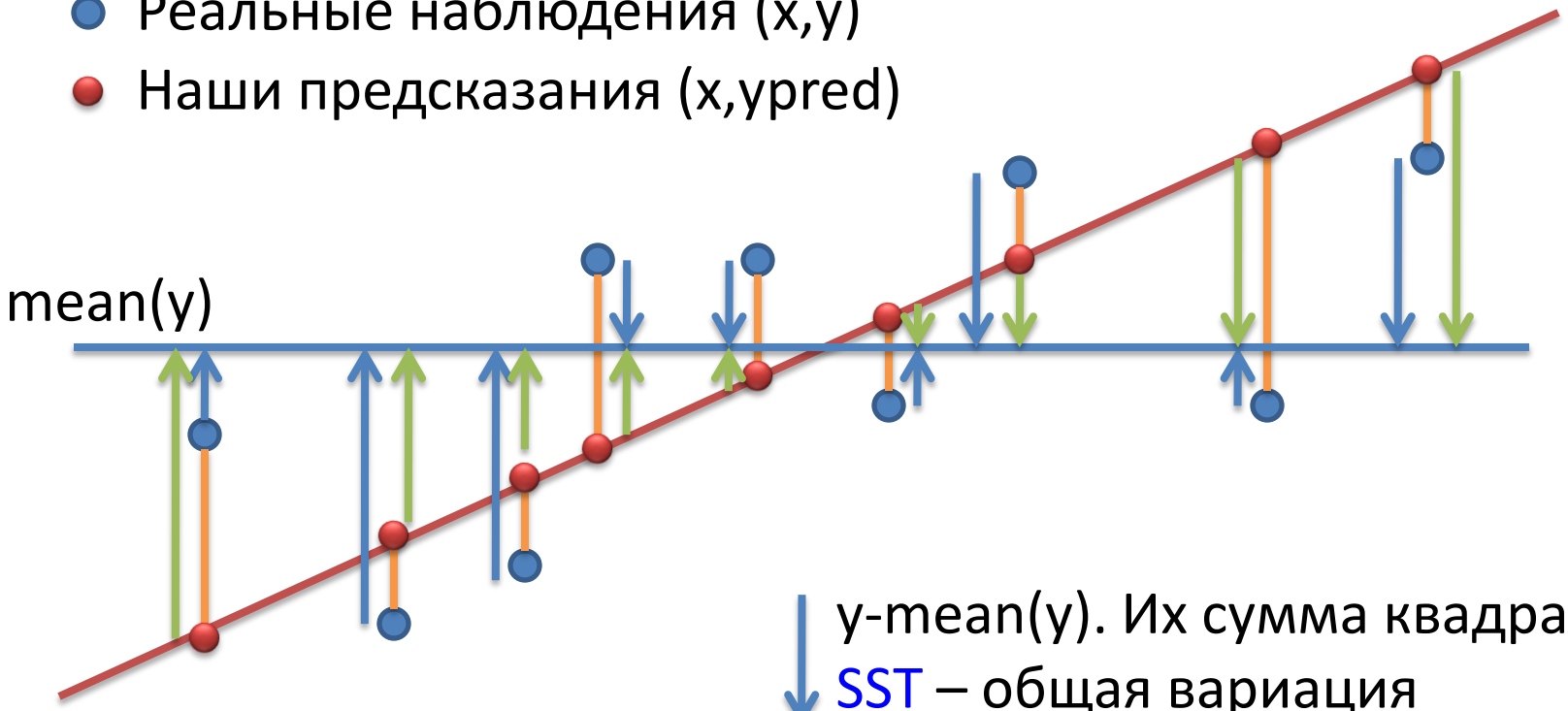


Figure 1: ANOVA : Fair fit

Wikipedia:ANOVA

SST, SSX, SSE

- Реальные наблюдения (x,y)
- Наши предсказания (x,ypred)



- R^2=SSX/SST
- F=(SSX)/
- /(SSE/(npoints-nlevels))

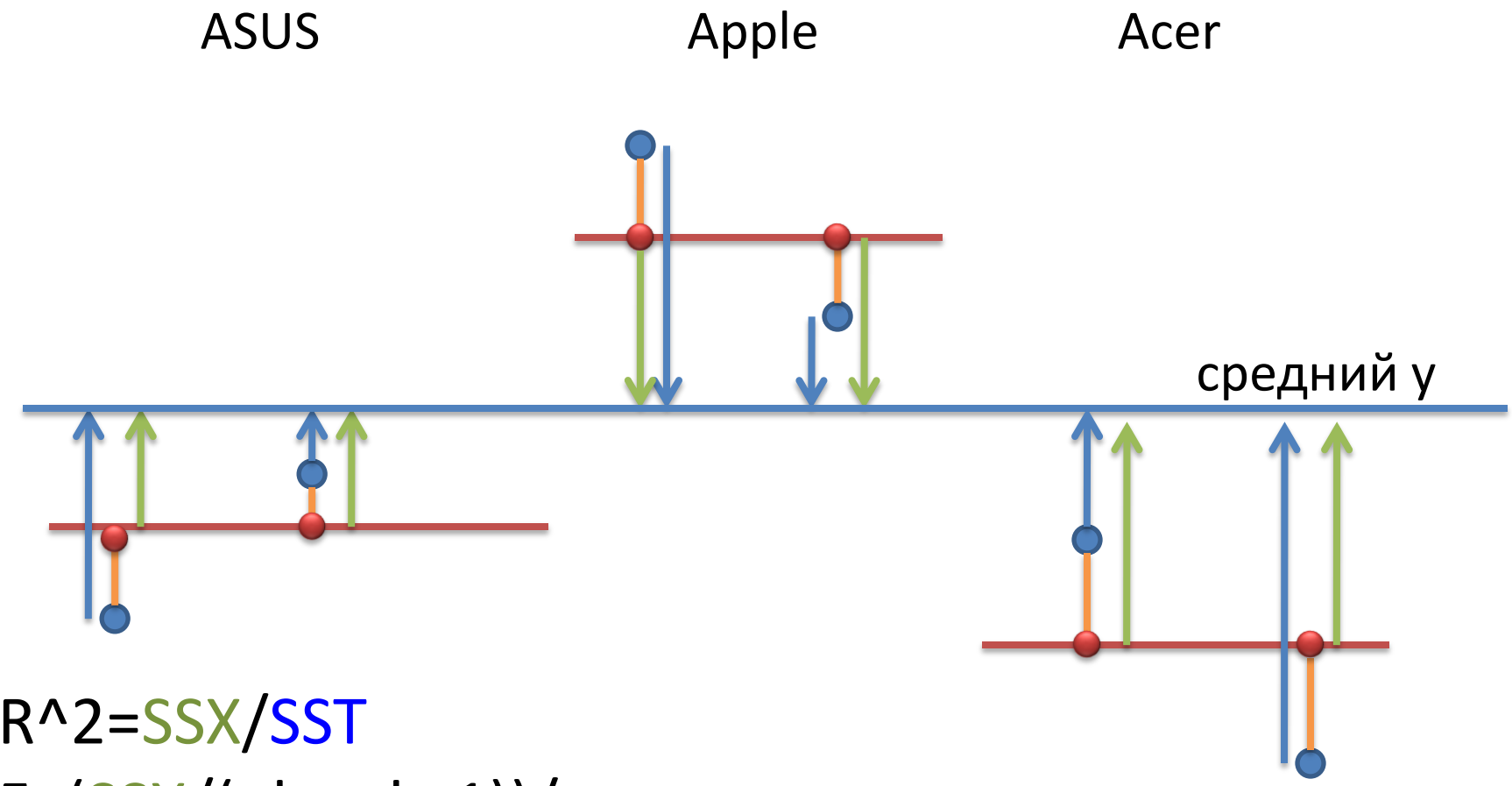
y-mean(y). Их сумма квадратов
SST – общая вариация

y-ypred. Ошибки.
SSE – необъясненная вариация

ypred-mean(y). Ошибки.
SSX – объясненная вариация

SST, SSX, SSE для факторного x

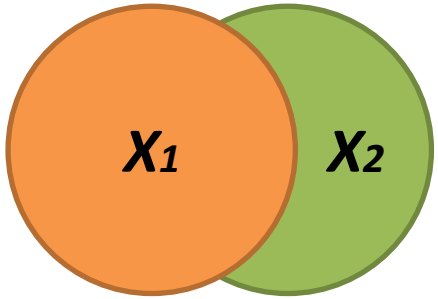
- Если x – фактор, то *ypred* – просто среднее по группе.



- R^2=SSX/SST
- F=(SSX/(nlevels-1))/
- /(SSE/(npoints-nlevels))

ANOVA

```
> l_MC=lm(Price_RUR ~ Manufacturer + Color, data=laptop)
> anova(l_MC)
…
                Df   Sum Sq  Mean Sq F value  Pr(>F)
Manufacturer     3 2.89e+10 9.64e+09    71.1 < 2e-16 ***
Color           14 5.12e+09 3.65e+08     2.7 0.00097 ***
Residuals      288 3.90e+10 1.36e+08

> l_CM=lm(Price_RUR ~ Color + Manufacturer, data=laptop)
> anova(l_CM)
…
                Df   Sum Sq  Mean Sq F value Pr(>F)
Color           14 1.79e+10 1.28e+09    9.45 <2e-16 ***
Manufacturer     3 1.61e+10 5.37e+09   39.61 <2e-16 ***
Residuals      288 3.90e+10 1.36e+08
```

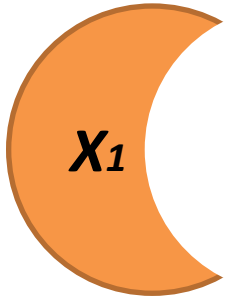


Важен порядок слагаемых! Если предикторы скоррелированы, то часть вариации может объясняться как первой, так и второй переменной. В стандартной ANOVA первая переменная берет на себя пересечение вариаций, следующая – то, что осталось

http://stats.stackexchange.com/questions/20002/regression-vs-anova-discrepancy

summary vs ANOVA

```
> summary(l_MC)
…
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)          19859       1716   11.58  < 2e-16 ***
ManufacturerApple    39827       3734   10.67  < 2e-16 ***
...
Colorblue            -5318       4567   -1.16  0.24519
...
Colorsilver           7591       1990    3.82  0.00017 ***
> summary(l_CM)
Estimate Std. Error t value Pr(>|t|)
(Intercept)          19859       1716   11.58  < 2e-16 ***
Colorblue            -5318       4567   -1.16  0.24519
...
Colorsilver           7591       1990    3.82  0.00017 ***
...
ManufacturerApple    39827       3734   10.67  < 2e-16 ***
```



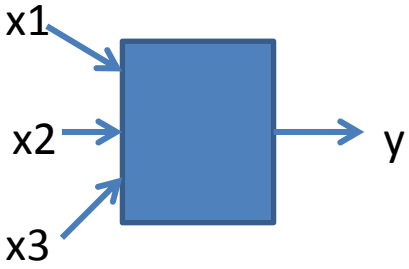
Для summary не важен порядок слагаемых. Для каждой переменной Xi t-test-ом оценивает, отличен ли её коэффициент от 0, по соотношению необъясненной и объясненной этим Xi вариации при данных значениях других X.

http://stats.stackexchange.com/questions/20002/regression-vs-anova-discrepancy

Для чего нужны линейные модели?

Входные данные

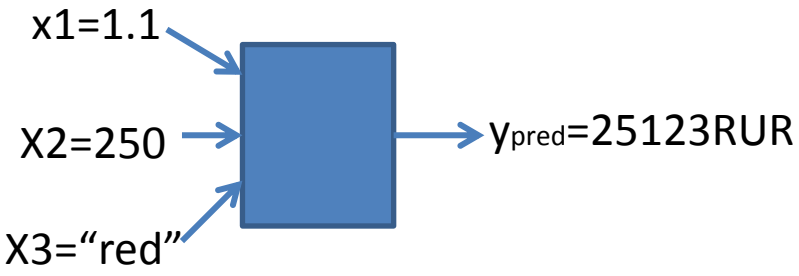
y	x1	x2	x3



Значимость каждой переменной:

Предсказание *y* по *x*

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   9503.178   1733.057   5.483 9.10e-08 ***
Memory_Gb     6232.204    421.098  14.800  < 2e-16 ***
HDD_Gb         -26.604      3.275  -8.123 1.34e-14 ***
Colorblue    -2496.492   4012.744  -0.622   0.5343
…
Colorred      1685.698   2736.167   0.616   0.5383
Colorsilver   8617.956   1679.963   5.130 5.33e-07 ***
…
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10330 on 289 degrees of freedom
Multiple R-squared: 0.578,      Adjusted R-squared: 0.5547
F-statistic: 24.74 on 16 and 289 DF,  p-value: < 2.2e-16
```



predict

```
> L3 = lm(formula = Price_RUR ~ HDD_Gb +
  HDD_type + HDD_Gb:HDD_type, data = laptop)
> newlaptops=data.frame( HDD_Gb=c(200, 1000,
  500), HDD_type=c("SSD", "HDD", "HDD"))
> newlaptops
  HDD_Gb HDD_type
1    200      SSD
2   1000      HDD
3    500      HDD

> predict( L3, newlaptops )
       1        2        3
50290.44 29700.06 21065.29
```

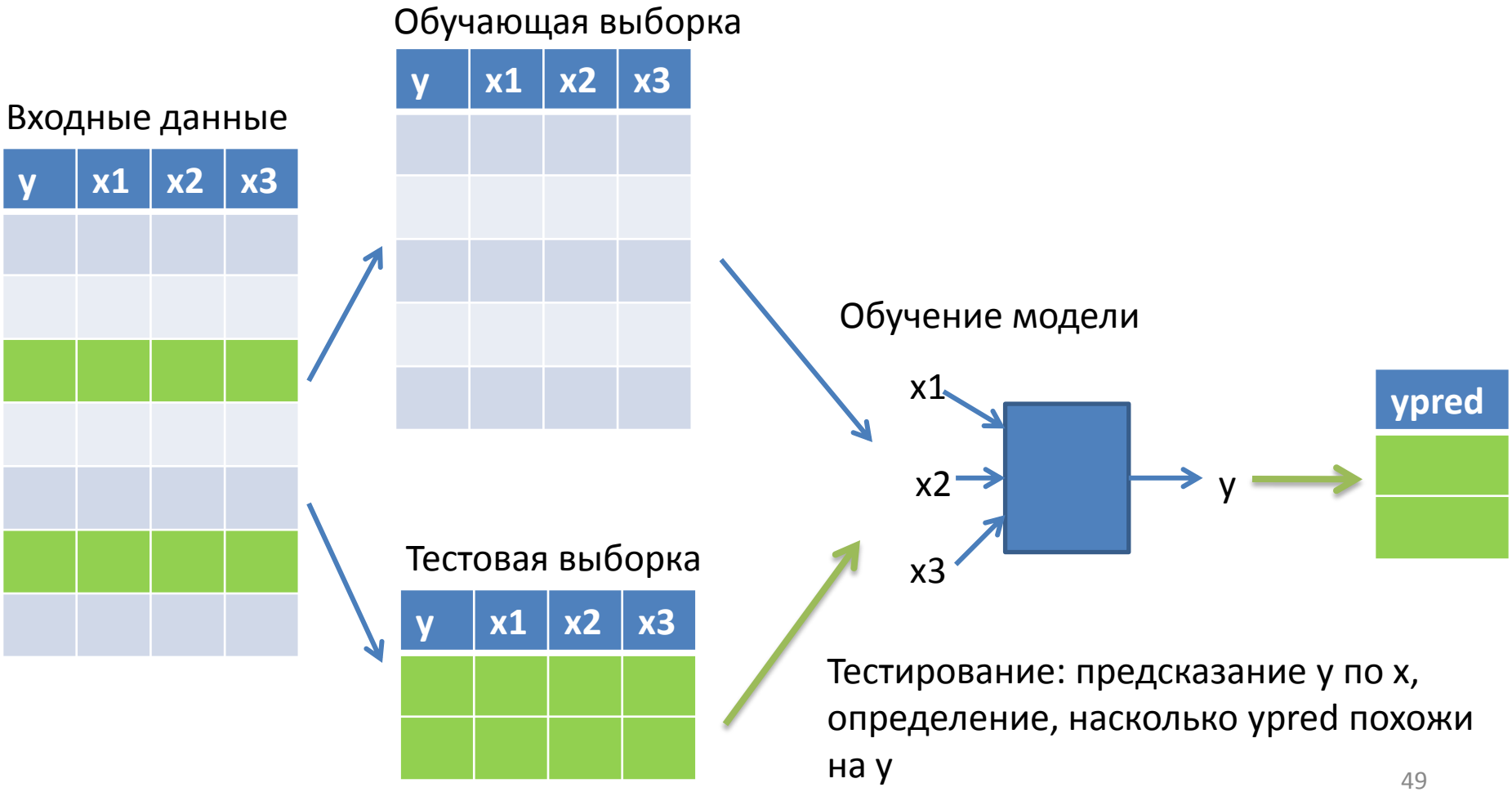
Модель

Dataframe с x-координатами новых точек, для которых делается предсказание y. Названия колонок должны соответствовать предикторам модели

Вектор предсказанных значений y

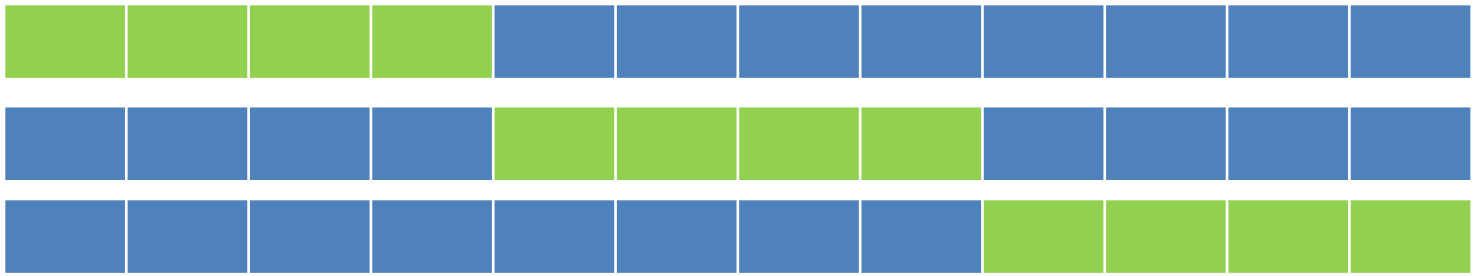
Кросс-валидация

- Для обучения модели и для её тестирования используются разные образцы (=строки в таблице).

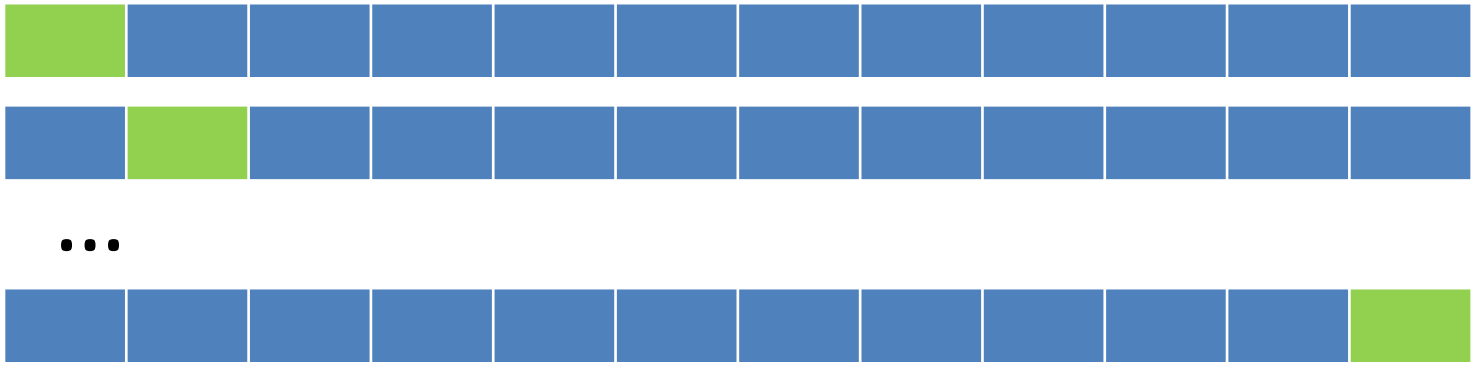


Методы кросс-валидации

- K-fold



- leave one out



Кросс-валидация. Пример

#(!) Устанавка дополнительного пакета

> install.packages("DAAG")

#Его подключение

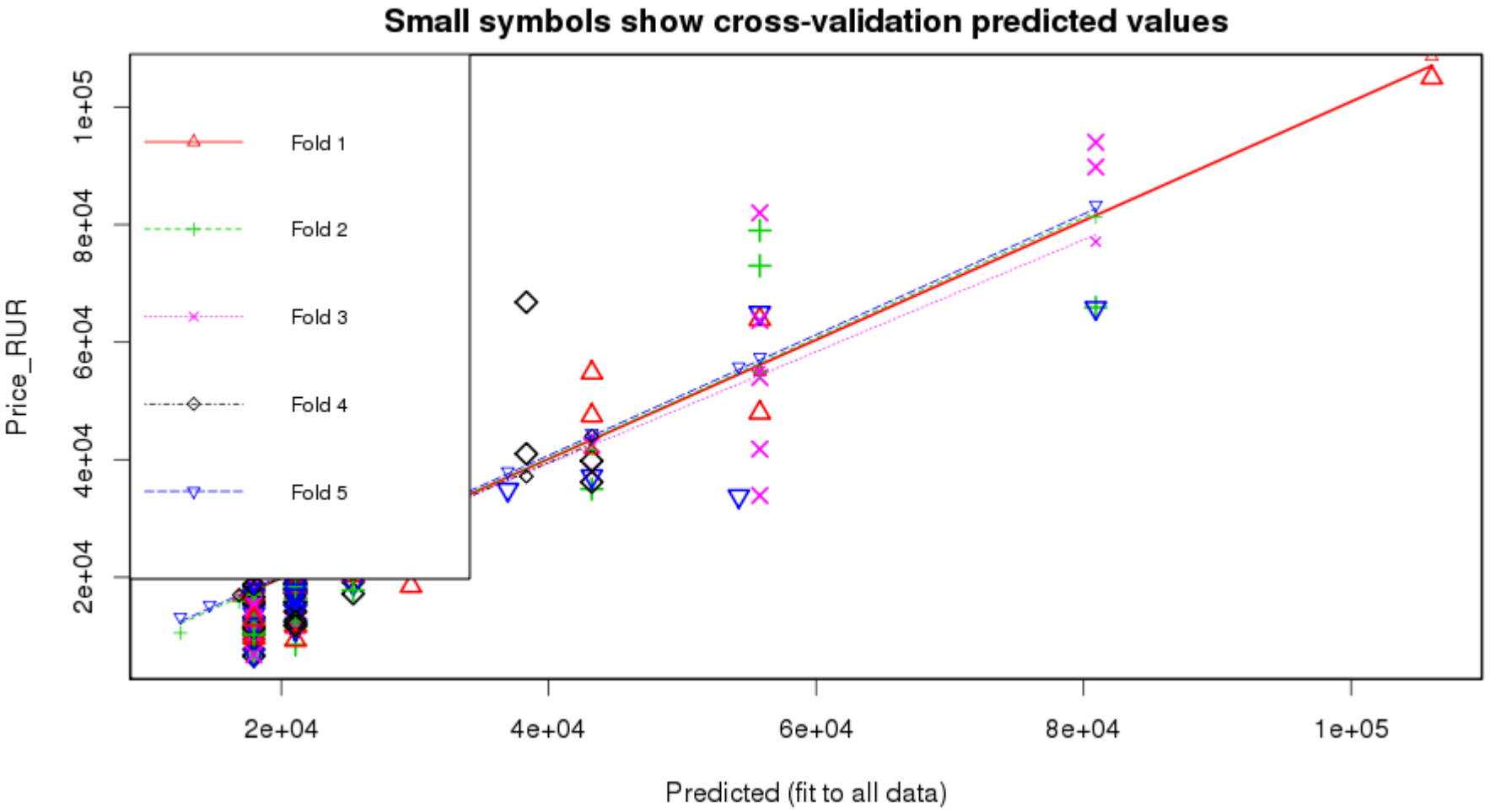
> library('DAAG‘) или > library(DAAG)

> cv.lm(laptop, L3, m=5)

Дополнительный аргумент – функция cost, по умолчанию:

cost=function(y, ypred) { mean ((y-ypred)^2) }

Кросс-валидация. Пример

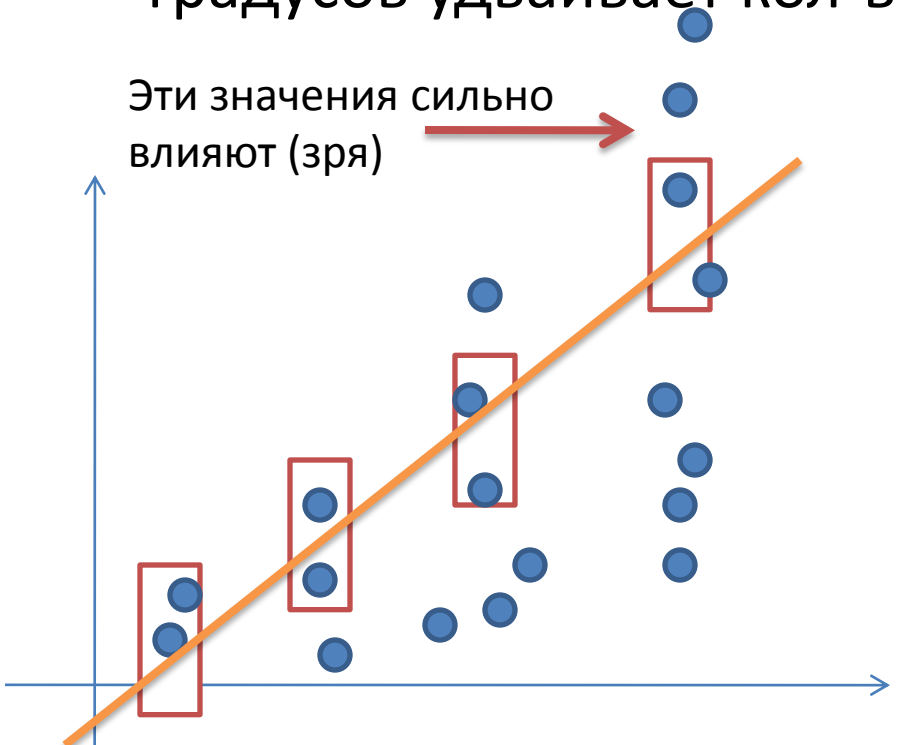


glm – обобщенные линейные модели

- Мотивация: иногда линейные модели не только не точны, но и по смыслу не подходят.
- Пример 1: как зависит количество людей на пляже от температуры
  - $f(-20)=-100$ человек?
- Пример 2: как зависит решение одного человека идти на пляж от температуры
  - p in [0,1]

glm: link function; var(mean)

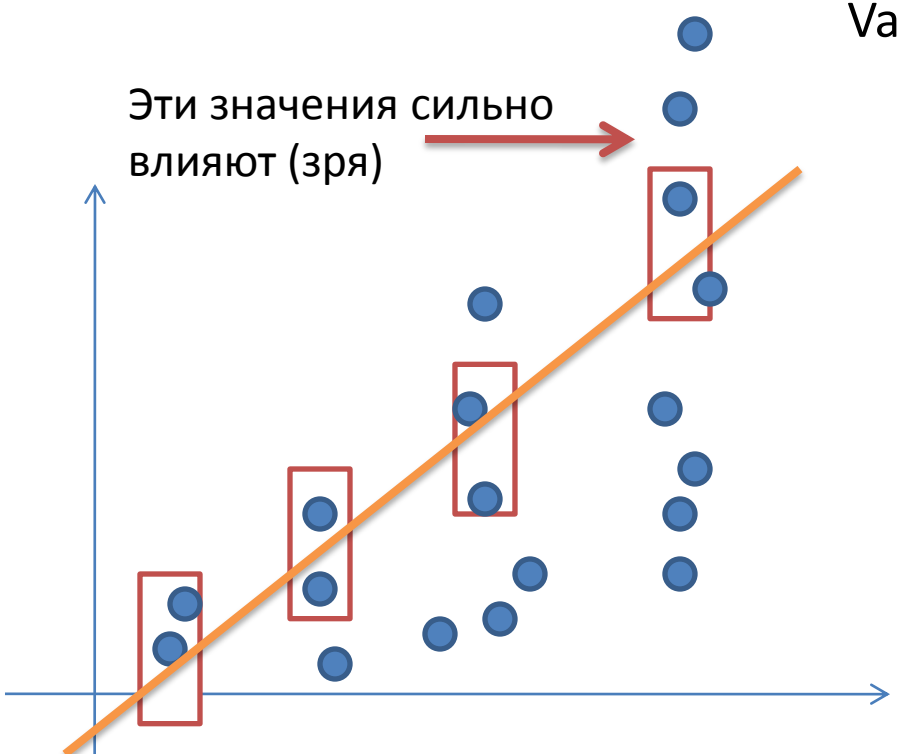
- 2 проблемы:
  - область определения y не соответствует области определения взвешенной суммы Xi
  - в разных областях y имеет разную дисперсию
- Например, пусть увеличение температуры на 5 градусов удваивает кол-во людей на пляже

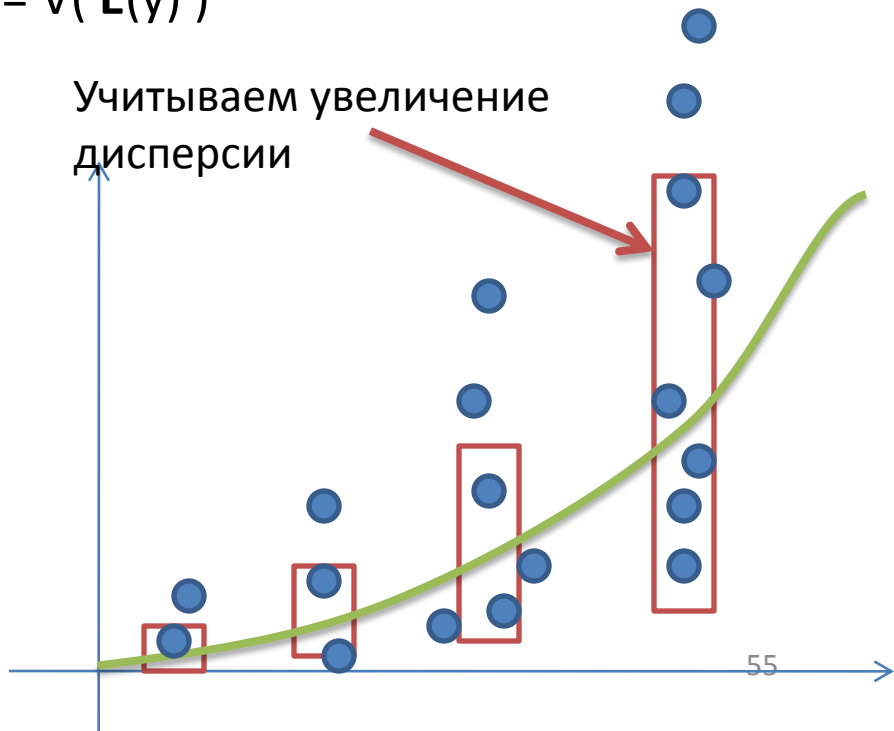


glm: link function; var(mean)

- Решения:
  - область определения ***y*** не соответствует области определения взвешенной суммы ***Xi***
    - link function g: $E(y) = g(\alpha + \beta x)$
  - в разных областях y имеет разную дисперсию
    - Предполагаем некоторую зависимость дисперсии от среднего

$$Var(y) = V(\mathbf{E}(y))$$





Логистическая регрессия

- Зависимая переменная принимает два значения (болен-здоров, жив-мертв, …)

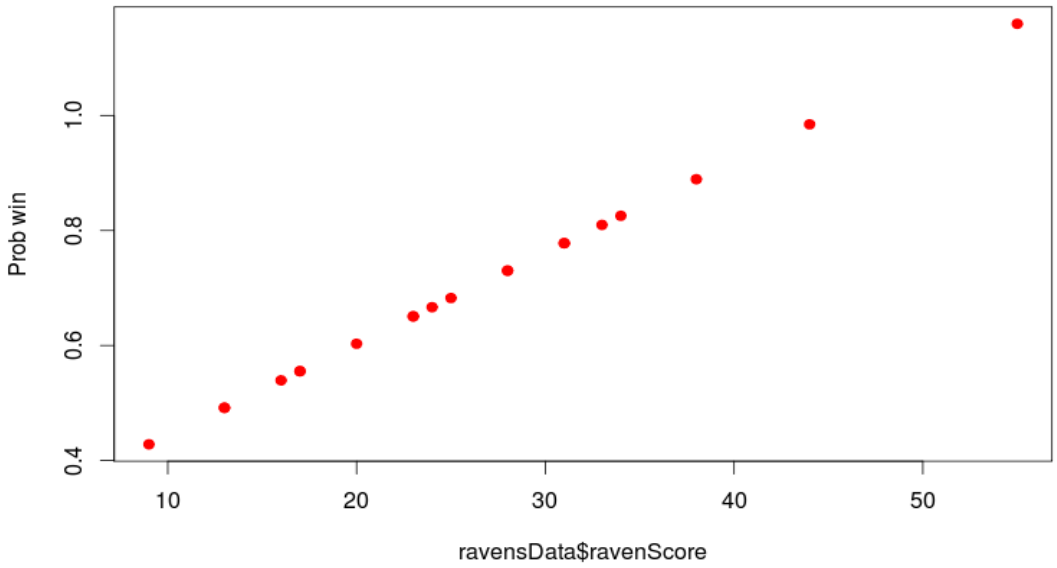
  Пример: таблица выигрышей команды. Хотим вычислять **вероятность** выигрыша в зависимости от количества очков

```
> load("ravensData.rda")
> head(ravensData)
  ravenWinNum ravenWin ravenScore opponentScore
1           1        W         24             9
2           1        W         38            35
3           1        W         28            13
4           1        W         34            31
5           1        W         44            13
6           0        L         23            24
```

Линейная регрессия не подходит

```
> lmRav<-
    lm(ravensData$ravenWinNum~ravensData$ravenScore)
> plot(ravensData$ravenScore, lmRav$fitted, pch=19, col='red',
    ylab="Prob win")
```

Для некоторых значений «предсказание» больше 1



Логистическая регрессия. link function

- Цель: превратить взвешенную сумму предикторов (любое число) в число из [0,1]

$$\log\left(\frac{Pr(RW_i|RS_i, b_0, b_1)}{1-Pr(RW_i|RS_i, b_{0,} b_1)}\right)=b_0+b_1 RS_i$$

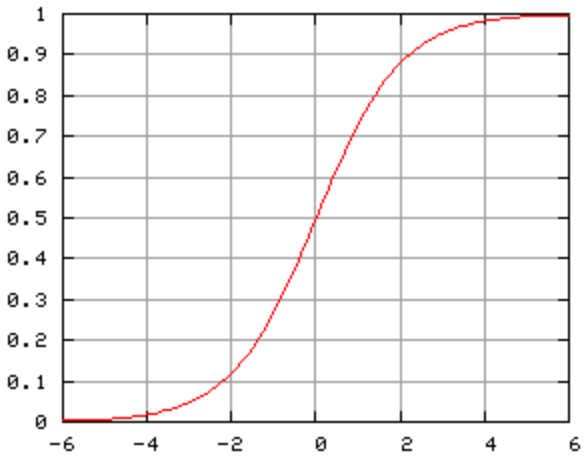
где RW_i - выигрыш (0 или 1)

RS_i - количесво очков

link function g:

g(p)=log(p / (1-p))

График g^-1



Логистическая регрессия

```
> logReg<-glm(
ravensData$ravenWinNum~ravensData$ravenScore,
family="binomial")
> logReg

Call:  glm(formula = ravensData$ravenWinNum ~
ravensData$ravenScore,
    family = "binomial")
Coefficients:
          (Intercept)  ravensData$ravenScore
              -1.6800                 0.1066
Degrees of Freedom: 19 Total (i.e. Null);  18
Residual
Null Deviance:      24.43
Residual Deviance: 20.89        AIC: 24.89
```

Предсказанные вероятности

```
> plot(ravensData$ravenScore, ravensData$ravenWinNum,
    pch=19, col='blue')
> points(ravensData$ravenScore, logReg$fitted, pch=19,
    col='red')
```