Does radon activity depend on floor?

Several ways to answer the same question:

- Hypothesis testing (t-test)
- Correlation
- Linear regression model

[1] Quinn G.P. and Keough M.J. Experimental design and data analysis for biologists. Cambridge University Press, 2002

Linear relationship between two variables: Pearson's correlation

(sample) Standard Deviation of x_1: $\sigma_{x1} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_{1i} - \bar{x_1})^2}$

(sample) Standard Deviation of x_2: $\sigma_{x2} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_{2i} - \bar{x_2})^2}$

(sample) Covariance: $\sigma_{x1,x2} = \frac{\sum_{i=1}^{N}(x_{1i}-\bar{x_1})(x_{2i}-\bar{x_2})}{N-1}$

(sample) Correlation: $\rho_{x1,x2} = \frac{\sigma_{x1,x2}}{\sigma_{x1}\cdot\sigma_{x2}}$

Note: The joint distribution of x_1 and x_2 should be a bivariate normal and their relationship should linear.

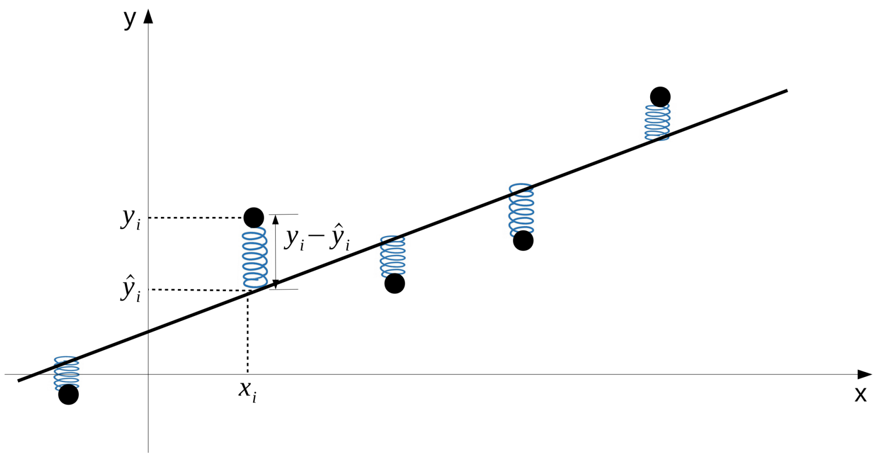
In [7]:

```
# radon <- read.table("http://www.stat.columbia.edu/~gelman/arm/examples/radon/
srrs2.dat", header=T, sep=",")

radon <- read.csv("../radon_colombia/srrs2.dat", header=T, sep=",")

radon <- radon[radon$floor < 9 , ] # remove floor 9

radon[ , "log_activity"] <- log( radon[ , "activity" ] + 1 )  # log transform a
ctivity

radon[1 : 2, ]
```

A data.frame: 2 × 26

	idnum	state	state2	stfips	zip	region	typebldg	floor	room	basement	⋯	startdt	stopdt	activity	pcterr	adjwt
	<int>	<fct>	<fct>	<int>	<int>	<int>	<int>	<int>	<int>	<fct>	⋯	<int>	<int>	<dbl>	<dbl>	<dbl>
1	1	AZ	AZ	4	85920	1	1	1	2	N	⋯	112987	120287	0.3	0	136.0610
3	3	AZ	AZ	4	85924	1	1	1	3	N	⋯	70788	70788	0.5	0	150.2451

Linear regression model



y = response variable, dependent variable, variable of interest

x = predictor variable, independent variable, covariate

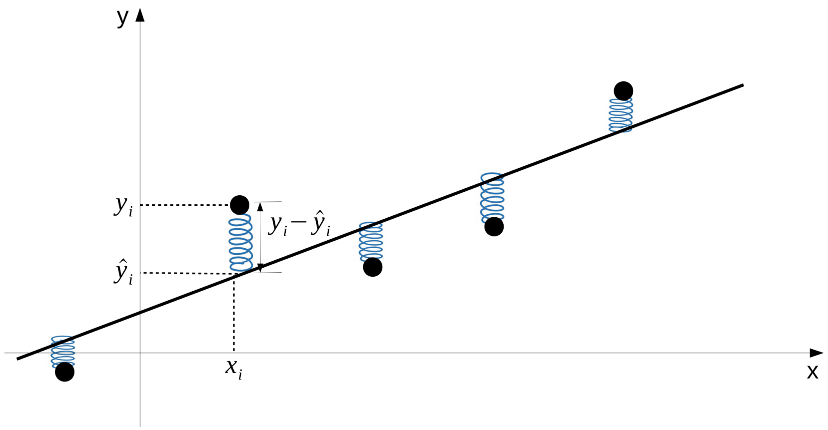
We expect that the predictor variable(s) may provide some explanation for the pattern we see in the response variable.

Three major purposes of linear regression analysis

1) To describe the linear relationship between y and x.

2) To determine how much of the variation (uncertainty) in y can be explained by the linear relationship with x and how much of this variation remains unexplained.

3) To predict new values of y from new values of x.



Consider a set of n observations (x_i, y_i), i = 1 : n.

The regression line is

$$\hat{y}_i = \hat{y}(x_i) = \theta_1 + x_i\theta_2$$

The linear regression model is

$$y_i = \theta_1 + x_i\theta_2 + \epsilon_i$$

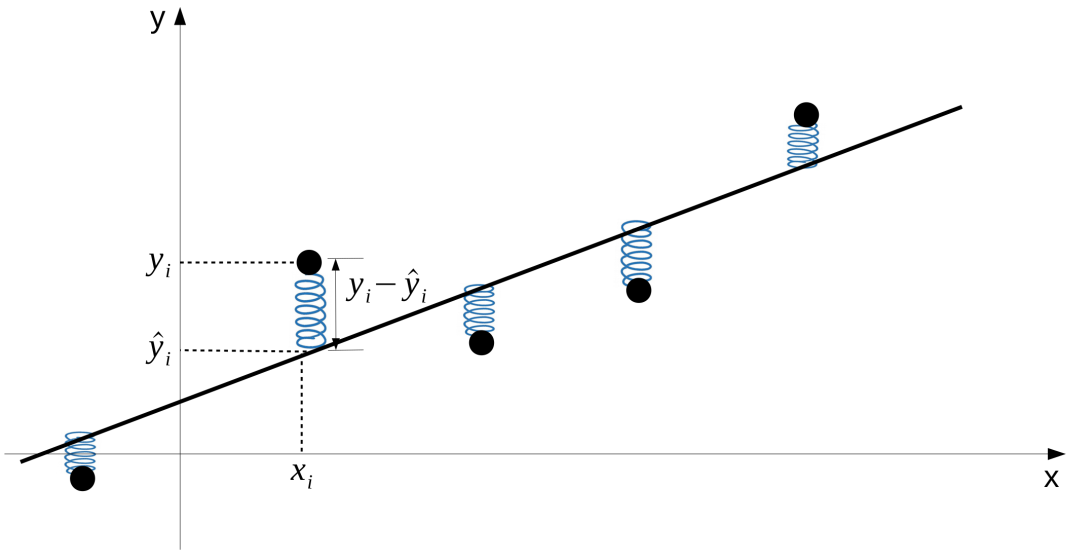
The error ϵ_i represents the part of y not explained by the model.

$\epsilon_i = y_i - \hat{y}_i$ (residuals)

We assume that the ϵ_i are i.i.d. and $\epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2)$.

$\hat{y}_i = \theta_1 + x_i\theta_2$

Two ways to estimate θ: Ordinary Least Squares (**OLS**) and Maximum Likelihood (**ML**).



OLS: minimizes the sum of the squares of the residuals $\epsilon_i = y_i - \hat{y}_i$ to find the model parameters θ

$J(\theta) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - \theta_1 - x_i\theta_2)^2$

$J(\theta) =$ energy, cost, objective function

In matricial form (n samples, 1 covariate):

- $\mathbf{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \dots & \dots \\ 1 & x_n \end{bmatrix}$ $\rightarrow$ design matrix (n x 2); $\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_n \end{bmatrix}$ $\rightarrow$ outcome (n x 1);

  $\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ $\rightarrow$ model parameters (2 x 1)

Hence: $\hat{\mathbf{y}} = \mathbf{X} \cdot \theta$

and the cost function is:
$J(\theta) = (\mathbf{y} - \mathbf{X}\theta)^T(\mathbf{y} - \mathbf{X}\theta) = \sum_{i=1}^{n}(y_i - \mathbf{x_i}\theta)^2 = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

where $\mathbf{x_i} = [1, x_i]$

Find the minimum by setting $\frac{\partial J(\theta)}{\partial \theta} = 0$

$$J(\theta) = (\mathbf{y} - \mathbf{X}\theta)^T(\mathbf{y} - \mathbf{X}\theta)$$

Given that $(AB)^T = B^T A^T$ and that if x and y are column (or row) vectors, than $x^T y$ is a real number and is equal to its transpose, thus $x^T y = (x^T y)^T = y^T x$:

$$\frac{\partial J(\theta)}{\partial \theta} = \frac{\partial}{\partial \theta}\left[y^T y + \theta^T X^T X\theta - 2y^T X\theta\right] = 0 + 2X^T X\theta - 2X^T y = 0$$

$$\Rightarrow 2X^T X\theta = 2X^T y$$

$$\Rightarrow \hat{\theta}_{OLS} = (X^T X)^{-1} X^T y$$

In [12]:

```
index_MN <- radon$state == "MN"

x <- radon[ index_MN, "floor" ]

data.frame( const=1, floor=x )[ 1 : 5, ]
```

A data.frame: 5 × 2

const	floor
<dbl>	<dbl>
1	1
1	0
1	0
1	0
1	0

In [13]:

```
y <- radon[ index_MN, "log_activity" ]

y[ 1 : 5 ]
```

1.16315080980568 1.16315080980568 1.3609765531356 0.693147180559945 1.41098697371026

The lm function in R uses OLS

In [14]:

```r
lm_pooled <- lm(y ~ x)

summary( lm_pooled )
```

```
Call:
lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.51801 -0.45017 -0.02409  0.40158  2.28257 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.61332    0.02173  74.242  < 2e-16 ***
x           -0.40165    0.05326  -7.542 1.12e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6014 on 917 degrees of freedom
Multiple R-squared:  0.0584,	Adjusted R-squared:  0.05737 
F-statistic: 56.88 on 1 and 917 DF,  p-value: 1.116e-13
```

One-sample Student's t-test

$\hat{\theta}$ = Estimate of slope; μ_θ = True value of $\hat{\theta}$

- $H_0 : \mu_\theta = 0$
- $H_1 : \mu_\theta \neq 0$

$t = \frac{\hat{\theta} - 0}{\hat{\sigma_\theta} / \sqrt{n}}$

where $\hat{\sigma_\theta} / \sqrt{n}$ = standard error of $\hat{\theta}$; n is the number of samples.

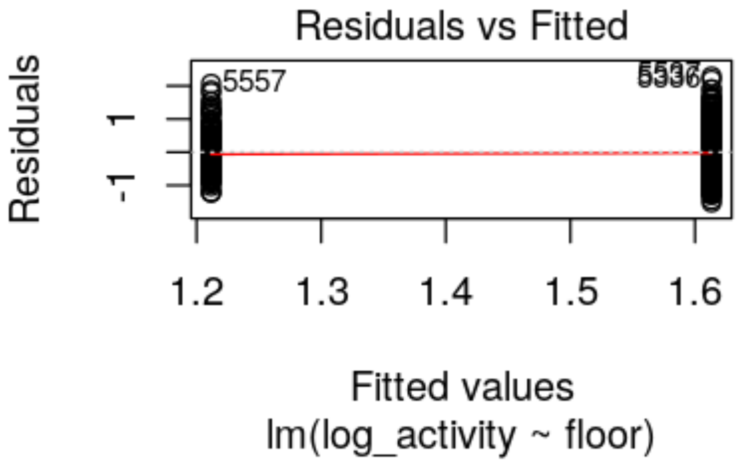
In [15]:

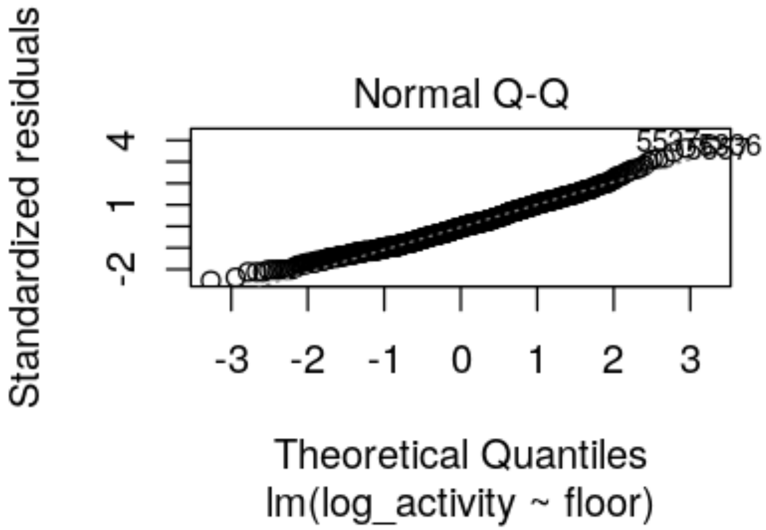
```
# OR:

index_MN <- radon$state=="MN"

lm_pooled <- lm( log_activity ~ floor, data = radon[index_MN, ] )

summary(lm_pooled)
```

```
Call:
lm(formula = log_activity ~ floor, data = radon[index_MN, ])

Residuals:
     Min       1Q   Median       3Q      Max 
-1.51801 -0.45017 -0.02409  0.40158  2.28257 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.61332    0.02173  74.242  < 2e-16 ***
floor       -0.40165    0.05326  -7.542 1.12e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6014 on 917 degrees of freedom
Multiple R-squared:  0.0584,	Adjusted R-squared:  0.05737 
F-statistic: 56.88 on 1 and 917 DF,  p-value: 1.116e-13
```

In [16]:

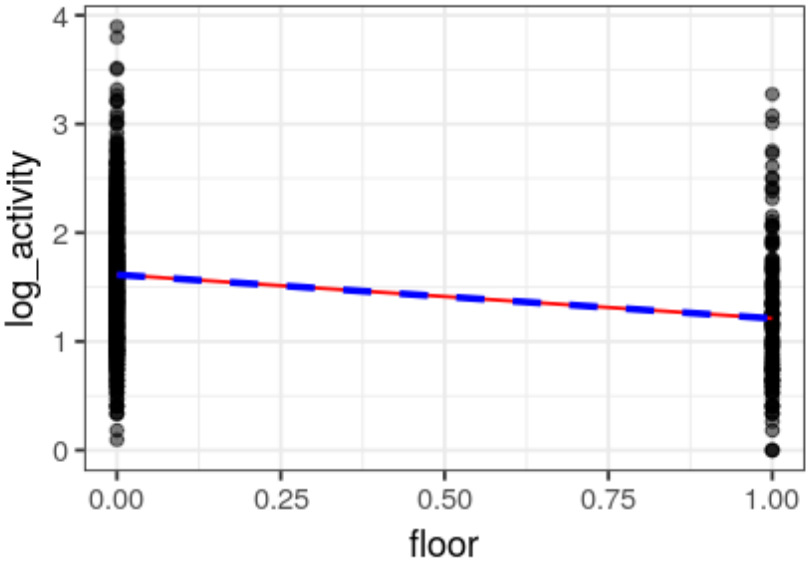
```
plot( lm_pooled )
```





In [17]:

```
pred_logact <- predict(lm_pooled, radon[index_MN, ]) # predict new data

predicted_df <- data.frame( pred_logact = pred_logact, floor = radon[index_MN,
"floor"] )

ggplot(data = radon[index_MN,], aes(x = floor, y = log_activity)) +

  geom_point(alpha=0.5) +

  geom_line(color='red',data = predicted_df, aes(x=floor, y=pred_logact)) +

  geom_smooth(method = "lm", se = FALSE, lty=2, color="blue") + #ggplot finds t
he same regression line

  theme_bw()
```

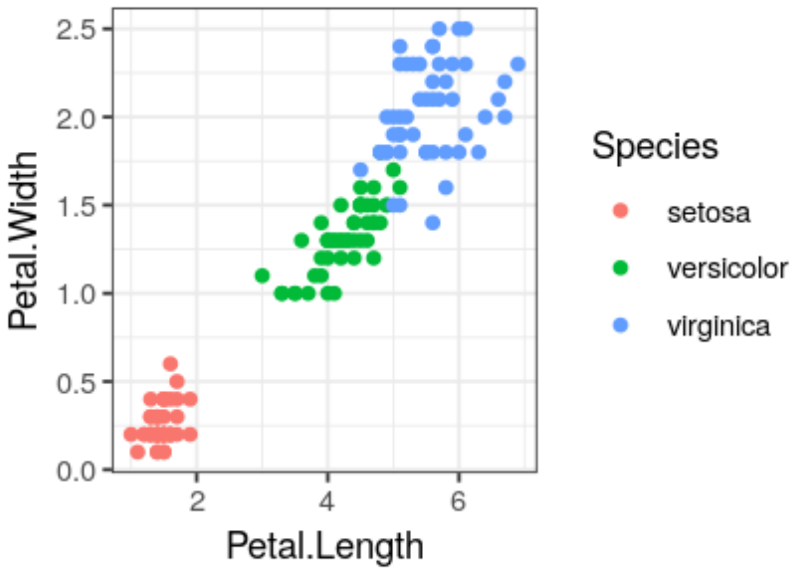


Exercise:

- Considering only the setosa species from the iris data set, compute the linear model of Petal.Width ~ Petal.Length
- Use the model to predict the Petal.Width of the virginica knowing their Petal.Length
- Visually check if the model fits well the setosa data
- Visually check if the model predicts well the virginica data

Solution:

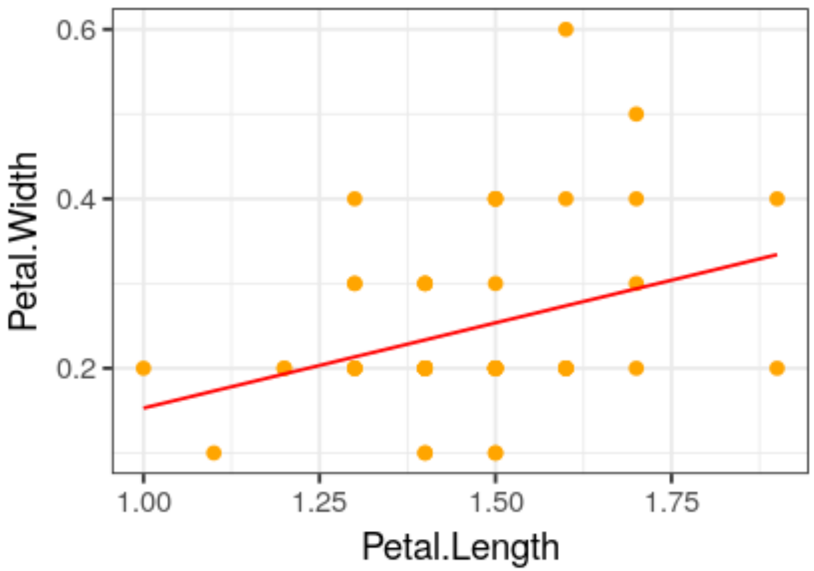
In [18]:
```
ggplot(data = iris, aes(x = Petal.Length, y = Petal.Width)) +
  geom_point(aes(col=Species)) +
  theme_bw()
```



In [19]:
```r
iris_setosa <- iris[ iris$Species == "setosa", ]

lm_setosa <- lm(Petal.Width ~ Petal.Length, data=iris_setosa)

summary(lm_setosa)
```

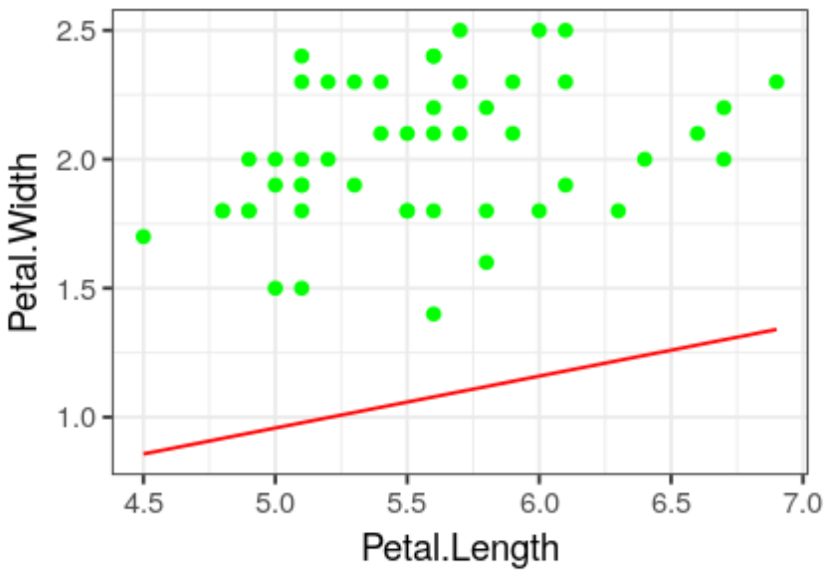
```
Call:
lm(formula = Petal.Width ~ Petal.Length, data = iris_setosa)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.15365 -0.05365 -0.03352  0.06632  0.32623 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)  
(Intercept)  -0.04822    0.12164  -0.396   0.6936  
Petal.Length  0.20125    0.08263   2.435   0.0186 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1005 on 48 degrees of freedom
Multiple R-squared:   0.11,	Adjusted R-squared:  0.09144 
F-statistic: 5.931 on 1 and 48 DF,  p-value: 0.01864
```

In [20]:
```
predict_setosa <- predict(lm_setosa, iris_setosa)

iris_setosa$predicted_Petal.Width <- predict_setosa

ggplot( data = iris_setosa, aes(x = Petal.Length, y = Petal.Width) ) +

  geom_point( color = 'orange' ) +

  geom_line( color = 'red', aes(x = Petal.Length, y = predicted_Petal.Width) )
+

  theme_bw()
```



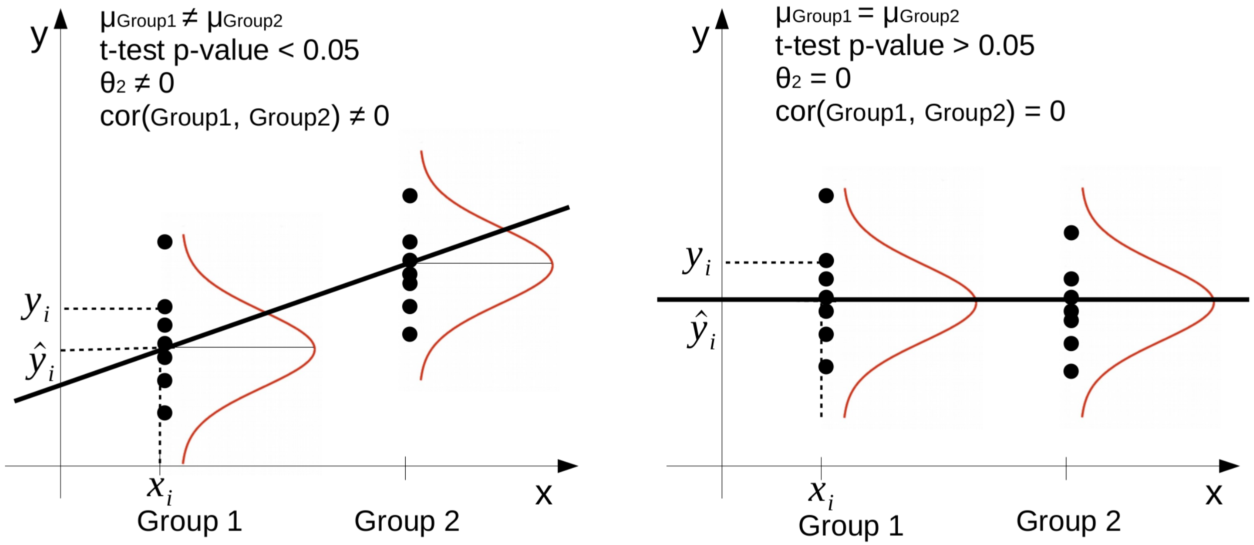
In [21]:

```
iris_virginica <- iris[iris$Species=="virginica",]

predict_virginica <- predict(lm_setosa, iris_virginica)

iris_virginica$predicted_Petal.Width <- predict_virginica

ggplot(data = iris_virginica, aes(x = Petal.Length, y = Petal.Width)) +

  geom_point(color='green') +

  geom_line(color='red',aes(x=Petal.Length, y=predicted_Petal.Width)) +

  theme_bw()
```



If $x = [\text{Group 1, Group 2}]$

$y = \theta_1 + \theta_2\, x + \varepsilon$
$E[y] = \theta_1 + \theta_2\, x$



In [22]:
```
index_MN_floor0 <- (radon$state == "MN") & (radon$floor == 0)

index_MN_floor1 <- (radon$state == "MN") & (radon$floor == 1)

y0 <- log(radon[index_MN_floor0,"activity"] + 1)

y1 <- log(radon[index_MN_floor1,"activity"] + 1)

t_test <- t.test(y0, y1)

t_test
```

```
	Welch Two Sample t-test

data:  y0 and y1
t = 7.1247, df = 206.35, p-value = 1.711e-11
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.2905067 0.5127922
sample estimates:
mean of x mean of y 
 1.613323  1.211673 
```

In [23]:

```
index_MN_floor01 <- (radon$state == "MN") & (radon$floor %in% c(0,1))

x <- radon[index_MN_floor01,"floor"]

y <- log(radon[index_MN_floor01,"activity"] + 1)

lm_01 <- lm(y~x)

summary(lm_01)
```

```
Call:
lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.51801 -0.45017 -0.02409  0.40158  2.28257 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.61332    0.02173  74.242  < 2e-16 ***
x           -0.40165    0.05326  -7.542 1.12e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6014 on 917 degrees of freedom
Multiple R-squared:  0.0584,	Adjusted R-squared:  0.05737 
F-statistic: 56.88 on 1 and 917 DF,  p-value: 1.116e-13
```

In [24]:

```
index_MN_floor01 <- (radon$state == "MN") & (radon$floor %in% c(0,1))

x <- radon[index_MN_floor01,"floor"]

y <- log(radon[index_MN_floor01,"activity"] + 1)

cor(x,y)
```

-0.241662618997735

If we scale x and y, then cor = θ_2

In [25]:

```r
index_MN_floor01 <- (radon$state == "MN") & (radon$floor %in% c(0,1))

x <- scale(radon[index_MN_floor01,"floor"])

y <- scale(log(radon[index_MN_floor01,"activity"] + 1))

print(paste("Pearson's cor = ", cor(scale(x),scale(y)),sep=""))

lm_01 <- lm(scale(y) ~ scale(x))

print("lm summary:")

summary(lm_01)
```

```
[1] "Pearson's cor = -0.241662618997735"
[1] "lm summary:"

Call:
lm(formula = scale(y) ~ scale(x))

Residuals:
    Min      1Q  Median      3Q     Max 
-2.4505 -0.7267 -0.0389  0.6483  3.6847 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -8.560e-16  3.203e-02   0.000        1    
scale(x)    -2.417e-01  3.204e-02  -7.542 1.12e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9709 on 917 degrees of freedom
Multiple R-squared:  0.0584,	Adjusted R-squared:  0.05737 
F-statistic: 56.88 on 1 and 917 DF,  p-value: 1.116e-13
```

Regression slope and Pearson's correlation

Minimize $J(\theta) = \sum_{i=1}^{n}(y_i - \theta_1 - \theta_2 x_i)^2$

expanding the square and deriving with respect to θ_1 and θ_2 we get:

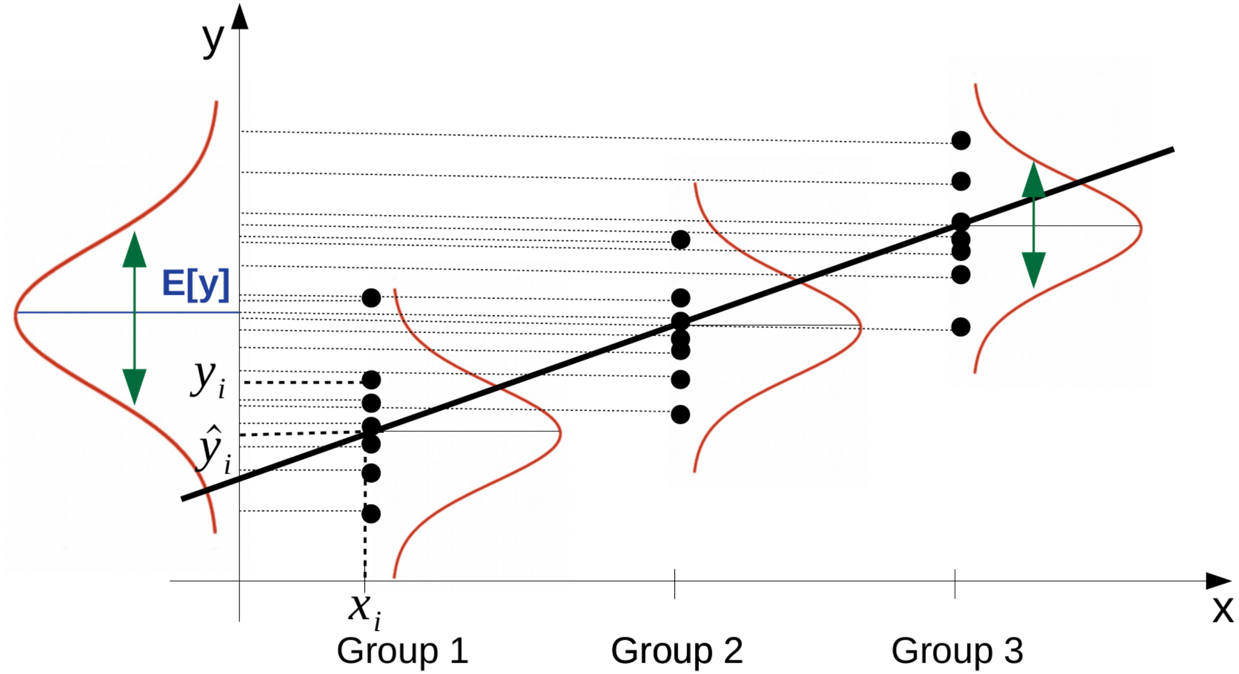
- $\hat{\theta}_1 = E[y] - \hat{\theta}_2 E[x]$
- $\hat{\theta}_2 = \frac{\sum_{i=1}^{n}(x_i - E[x])(y_i - E[y])}{\sum_{i=1}^{n}(x_i - E[x])^2} = \frac{\sigma_{x,y}}{\sigma_x^2}$

$\rho_{x,y} = \frac{\sigma_{x,y}}{\sigma_x \cdot \sigma_y}$

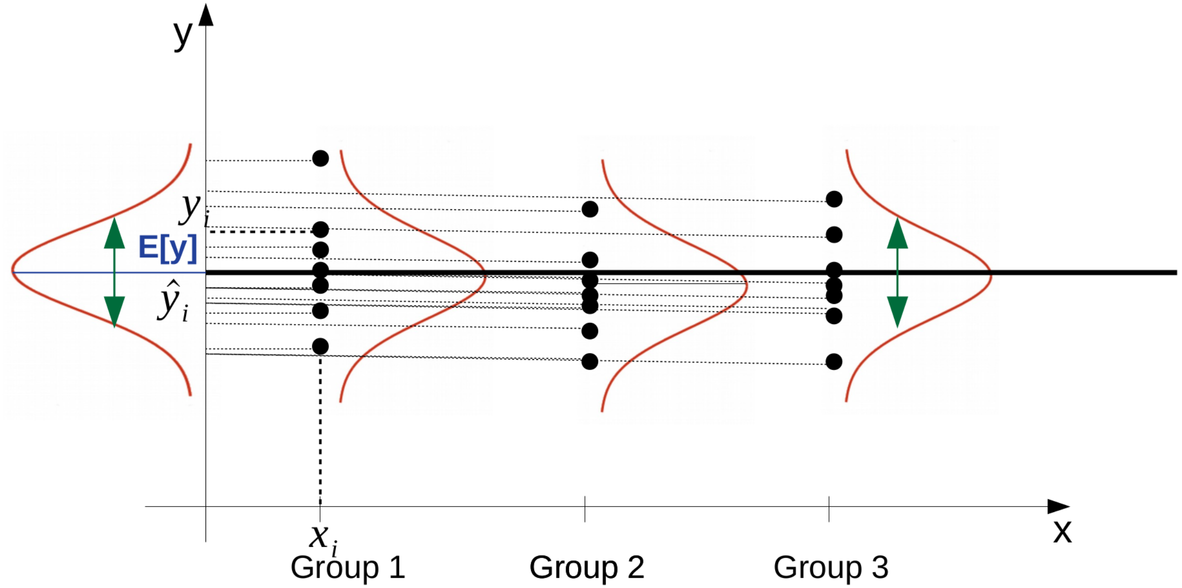
Then, $\hat{\theta}_2 = \rho_{x,y} \frac{\sigma_y}{\sigma_x}$

And if $\sigma_y = \sigma_x$, then $\hat{\theta}_2 = \rho_{x,y}$

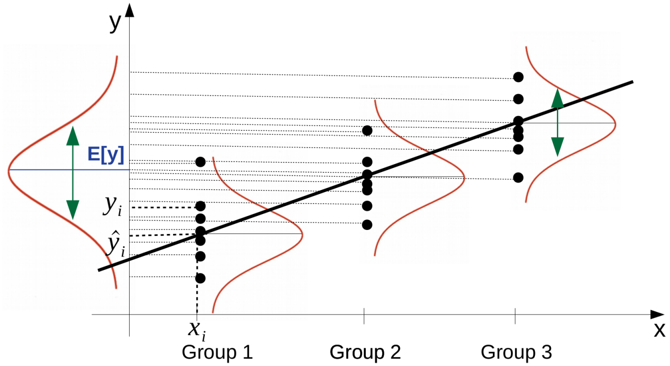
If $x = $[Group 1, Group 2, Group 3]



If x =[Group 1, Group 2, Group 3]



Analysis Of VAriance (ANOVA) $\hat{y}_i = \theta_1 + x_i\theta_2$



Source of variation	SS	df	MS	Expected mean square
Regression: Variability of y due to the relationship with x	$\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2$	Number of model parameters (2) - 1 1	$\frac{\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2}{1}$	$\sigma_\varepsilon^2 + \theta_2^2 \sum_{i=1}^{n}(x_i-\bar{x})^2$
Residual: Variability of y not explained by the relationship with x	$\sum_{i=1}^{n}(y_i-\hat{y}_i)^2$	Because we already estimated 2 param.: θ1 and θ2 $n-2$	$\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n-2}$	σ_ε^2 If homoscedasticity holds
Total: Total variability of y	$\sum_{i=1}^{n}(y_i-\bar{y})^2$	$n-1$ Because we already estimated $\bar{y}$		

Null hypothesis in regression:

$\hat{y}_i = \theta_1 + x_i\theta_2$

- $H_0 : \theta_2 = 0$
- $H_1 : \theta_2 \neq 0$

We can use the one sample t-test to test H_0.

If H_0 is true, both $MS_{Regression}$ and $MS_{Residual}$ estimate σ_ϵ^2.

Hence $\frac{MS_{Regression}}{MS_{Residual}} \leq 1$

If H_0 is NOT true, $\frac{MS_{Regression}}{MS_{Residual}} \geq 1$

ANOVA

Under the assumptions of: normality, independence and homoscedasticity, if H_0 is true:

$F = \frac{MS_{Regression}}{MS_{Residual}} \sim F-$distribution $\rightarrow$ p-value

Nested models

The F-test basically compares the fit to the data of a model that includes a slope term to the fit of a model that does not:

$y_i = \theta_1 + x_i\theta_2 + \epsilon_i \rightarrow$ its unexplained variance is $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = SS_{Residual}$

$y_i = \theta_1 + \epsilon_i \rightarrow$ its unexplained variance is $\sum_{i=1}^{n}(y_i - \bar{y})^2 = SS_{Total}$ because all predicted values $\hat{y}_i$ coincide with the intercept, which is $\bar{y}$.

$SS_{Total} - SS_{Residual} = SS_{Regression}$

and we can use $\frac{MS_{Regression}}{MS_{Residual}}$ to test if adding x to the model explains some variance.

Notes:

- $R^2 = \frac{SS_{Regession}}{SS_{Total}} = 1 - \frac{SS_{Residual}}{SS_Total}$
- If we use ML, we may do a Likelihood Ratio Test.

In [26]:

```
index_MN <- radon$state == "MN"

res.aov <- aov( log_activity ~ floor, data = radon[index_MN, ] )

# Summary of the analysis:

summary(res.aov)
```

```
             Df Sum Sq Mean Sq F value   Pr(>F)    
floor         1   20.6  20.573   56.88 1.12e-13 ***
Residuals   917  331.7   0.362                     
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

In [27]:
```
summary( lm( log_activity ~ floor, data=radon[index_MN, ] ) ) # The F-test is t
he same!
```

```
Call:
lm(formula = log_activity ~ floor, data = radon[index_MN, ])

Residuals:
     Min       1Q   Median       3Q      Max 
-1.51801 -0.45017 -0.02409  0.40158  2.28257 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.61332    0.02173  74.242  < 2e-16 ***
floor       -0.40165    0.05326  -7.542 1.12e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6014 on 917 degrees of freedom
Multiple R-squared:  0.0584,	Adjusted R-squared:  0.05737 
F-statistic: 56.88 on 1 and 917 DF,  p-value: 1.116e-13
```

In [28]:

```
model0 <- lm( log_activity ~ 1, data=radon[index_MN, ] )

model1 <- lm( log_activity ~ floor, data=radon[index_MN, ] )

anova(model0, model1)  # The F-test is the same!
```

A anova: 2 × 6

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
918	352.2736	NA	NA	NA	NA
917	331.7006	1	20.57307	56.87511	1.116116e-13

Exercise:

- Does the (log) activity of radon depends on the state?

Solution:

In [29]:
```
model_state <- lm(log_activity ~ state, data=radon)

summary(model_state)
```

```
Call:
lm(formula = log_activity ~ state, data = radon)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.8559 -0.4999 -0.0989  0.3970  4.1048 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.79558    0.01879   42.35   <2e-16 ***
stateIN      0.44693    0.02487   17.97   <2e-16 ***
stateMA      0.39748    0.02594   15.32   <2e-16 ***
stateMN      0.75087    0.02989   25.12   <2e-16 ***
stateMO      0.28907    0.02494   11.59   <2e-16 ***
stateND      1.06037    0.02577   41.14   <2e-16 ***
statePA      0.76189    0.02372   32.12   <2e-16 ***
stateR5      0.36100    0.02981   12.11   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7049 on 12482 degrees of freedom
Multiple R-squared:  0.1609,	Adjusted R-squared:  0.1605 
F-statistic:   342 on 7 and 12482 DF,  p-value: < 2.2e-16
```

Dummy variables

Categorical variables are converted to dummy variables.

model.matrix creates a design (or model) matrix, e.g. by expanding factors to a set of dummy variables.

In [30]:

```
state_f <- factor(radon$state)

dummies <- model.matrix(~state_f)

dummies
```

A matrix: 12490 × 8 of type dbl

(Intercept)	state_fIN	state_fMA	state_fMN	state_fMO	state_fND	state_fPA	state_fR5
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0

(Intercept)	state_fIN	state_fMA	state_fMN	state_fMO	state_fND	state_fPA	state_fR5
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1

In [31]:

```r
# One factor is excluded to avoid multicollinearity

print( length(unique(radon$state)) )

print( ncol(dummies) - 1 )

print( unique(radon$state) )

print( colnames(dummies) )
```

```
[1] 8
[1] 7
[1] AZ IN MA MN MO ND PA R5
Levels: AZ IN MA MN MO ND PA R5
[1] "(Intercept)" "state_fIN"   "state_fMA"   "state_fMN"   "state_fMO"
[6] "state_fND"   "state_fPA"   "state_fR5"
```

In [32]:

```
# we can have more than one covariate

summary( lm( log_activity ~ floor + state, data=radon ) )
```

```
Call:
lm(formula = log_activity ~ floor + state, data = radon)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.9164 -0.4840 -0.1048  0.3761  4.0670 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.21063    0.02361  51.271  < 2e-16 ***
floor       -0.42532    0.01537 -27.669  < 2e-16 ***
stateIN      0.23783    0.02530   9.400  < 2e-16 ***
stateMA      0.02023    0.02863   0.707 0.479851    
stateMN      0.40664    0.03157  12.880  < 2e-16 ***
stateMO      0.03513    0.02589   1.357 0.174768    
stateND      0.70581    0.02811  25.110  < 2e-16 ***
statePA      0.39171    0.02663  14.710  < 2e-16 ***
stateR5      0.10149    0.03042   3.336 0.000852 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6843 on 12481 degrees of freedom
Multiple R-squared:  0.2094,	Adjusted R-squared:  0.2089 
F-statistic: 413.3 on 8 and 12481 DF,  p-value: < 2.2e-16
```

Exercise:

- Compare the three nested models:
    - log_activity ~ 1
    - log_activity ~ floor
    - log_activity ~ floor + state
- Which model better describes the data?

Solution

In [33]:
```r
model0 <- lm(log_activity ~ 1, data=radon)

model1 <- lm(log_activity ~ floor, data=radon)

model2 <- lm(log_activity ~ floor + state, data=radon)

anova_df <- anova(model0,model1,model2)

anova_df[,"model"] <- c("Intercept", "floor", "floor+state")

anova_df
```

A anova: 3 × 7

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)	model
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
12489	7391.730	NA	NA	NA	NA	Intercept
12488	6464.799	1	926.9314	1979.7655	0.000000e+00	floor
12481	5843.637	7	621.1617	189.5278	4.397432e-268	floor+state

In [ ]: