Table of Contents


```
begin
        using PlutoUI    ,MLJ    , DataFrames    ,StatsPlots    ,StatsBase    ,RDatasets
    ,ScientificTypes
    PlutoUI.TableOfContents()
end
```

1. 生成线性回归数据

Regression-data-generated-from-noisy-linear-models

	x1	x2	x3	x4	x5	y
1	2.03465	-0.466263	1.18746	-0.255324	0.70891	-0.097129
2	0.0856653	-0.671893	0.145437	1.75428	-0.20806	0.568957
3	1.27709	0.41057	0.000103094	-1.34266	-1.08045	0.325458

```
begin

X1, y1 = make_regression(100, 5; noise=0.5, sparse=0.2, outliers=0.1)
dfRegression = DataFrame(X1)
dfRegression.y = y1
first(dfRegression, 3)
end
```

2. load data from RDatasets

> 使用数据,boston housing. 收集了波士顿地区房屋的一些特征和平均价格, 在线性回归中可以使用特征数值来预测房价. 特征数值作为预测变量(predict variables), 房价作为响应变量(responsive variable)

```
boston = dataset("MASS", "Boston");
```

names	scitypes	types
Crim	Continuous	Float64
Zn	Continuous	Float64
Indus	Continuous	Float64
Chas	Count	Int64
NOx	Continuous	Float64
Rm	Continuous	Float64
Age	Continuous	Float64
Dis	Continuous	Float64
Rad	Count	Int64
Tax	Count	Int64
⋮	⋮	⋮

4 rows omitted

```
schema(boston)  # using  ScientificTypes to show data type
```

	variable	mean	min	median	max	nmissing	eltype
1	:Crim	3.61352	0.00632	0.25651	88.9762	0	Float64
2	:Zn	11.3636	0.0	0.0	100.0	0	Float64
3	:Indus	11.1368	0.46	9.69	27.74	0	Float64
4	:Chas	0.06917	0	0.0	1	0	Int64
5	:NOx	0.554695	0.385	0.538	0.871	0	Float64
6	:Rm	6.28463	3.561	6.2085	8.78	0	Float64
7	:Age	68.5749	2.9	77.5	100.0	0	Float64
8	:Dis	3.79504	1.1296	3.20745	12.1265	0	Float64
9	:Rad	9.54941	1	5.0	24	0	Int64
10	:Tax	408.237	187	330.0	711	0	Int64
11	:PTRatio	18.4555	12.6	19.05	22.0	0	Float64
12	:Black	356.674	0.32	391.44	396.9	0	Float64
13	:LStat	12.6531	1.73	11.36	37.97	0	Float64
14	:MedV	22.5328	5.0	21.2	50.0	0	Float64

```
describe(boston)
```

3. using MLJ ensemble model

```
md"""## 3. using MLJ  ensemble model"""
```

MLJLinearModels.LinearRegressor

```
begin
    import MLJLinearModels
    MLJLinearModels.LinearRegressor
    LinearRegressor = @load LinearRegressor pkg=MLJLinearModels
end
```

```
[ import MLJLinearModels ✓ ] erbosity=0`.
```

```
data = coerce(boston, autotype(boston, :discrete_to_continuous)); #由于很多计算方法要用到浮点类型，执行这个操作将整数类型转化为浮点类型
```

([24.0, 21.6, 34.7, 33.4, 36.2, 28.7, 22.9, 27.1, 16.5, more ,11.9],

	Crim	
1	0.00632	18
2	0.02731	0.
3	0.02729	0.
4	0.03237	0.
5	0.06905	0.
6	0.02985	0.
7	0.08829	12
8	0.14455	12
9	0.21124	12
10	0.17004	12
	more	
506	0.04741	0.

```
y,X = data.MedV,select(data, Not(:MedV))
```

```
mdl = LinearRegressor(
        fit_intercept = true,
        solver = nothing)
```

```
mdl = LinearRegressor()  # 实例化线性模型
```

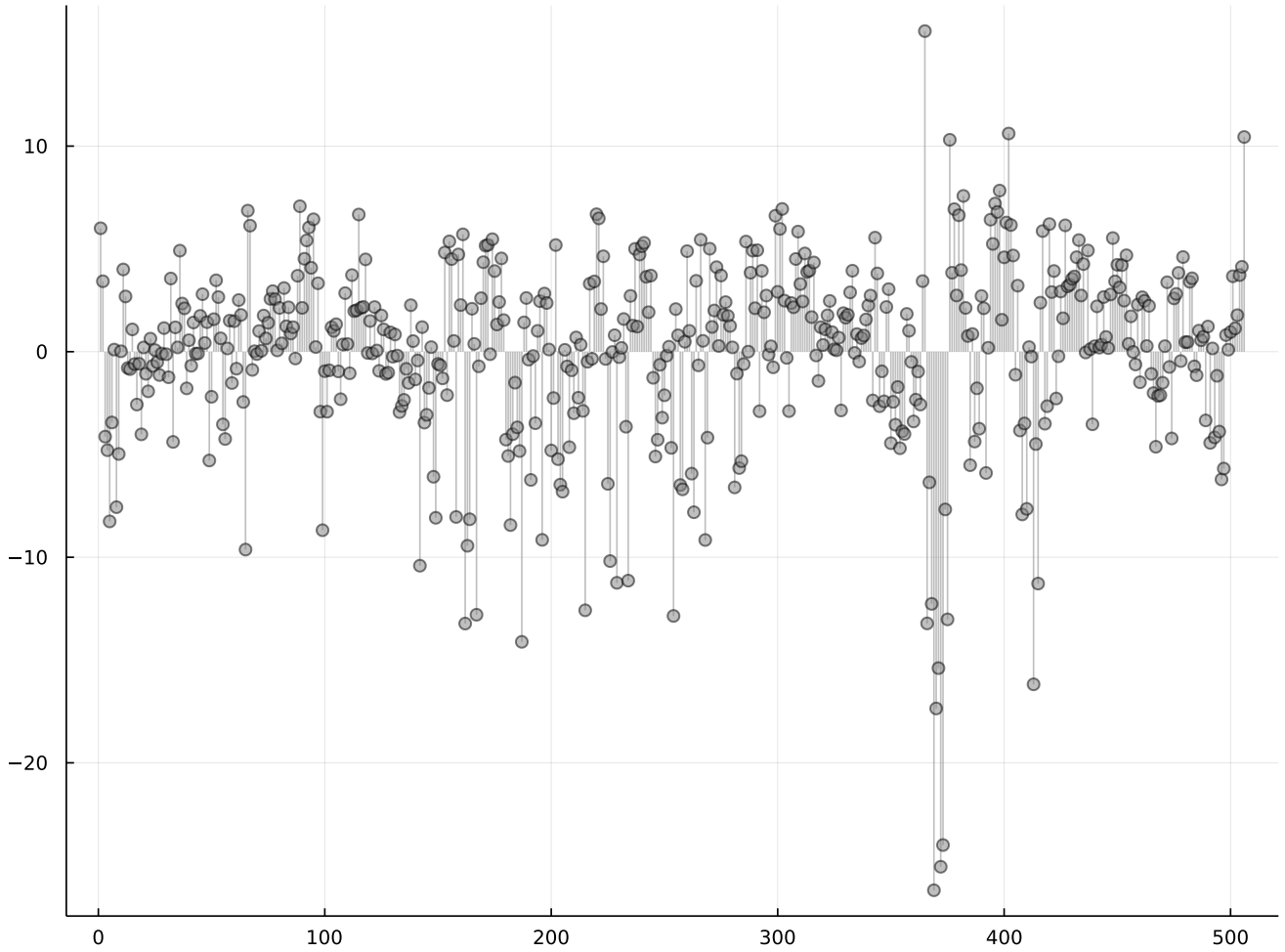
```
begin
    mach = machine(mdl, X, y)  #使用数据生成模型
    fit!(mach)

    fp = fitted_params(mach)
    coefs = fp.coefs
    intercept = fp.intercept
    for (name, val) in coefs
        println("$(rpad(name, 8)):  $(round(val, sigdigits=3))")
    end
    println("Intercept: $(round(intercept, sigdigits=3))")
end
```

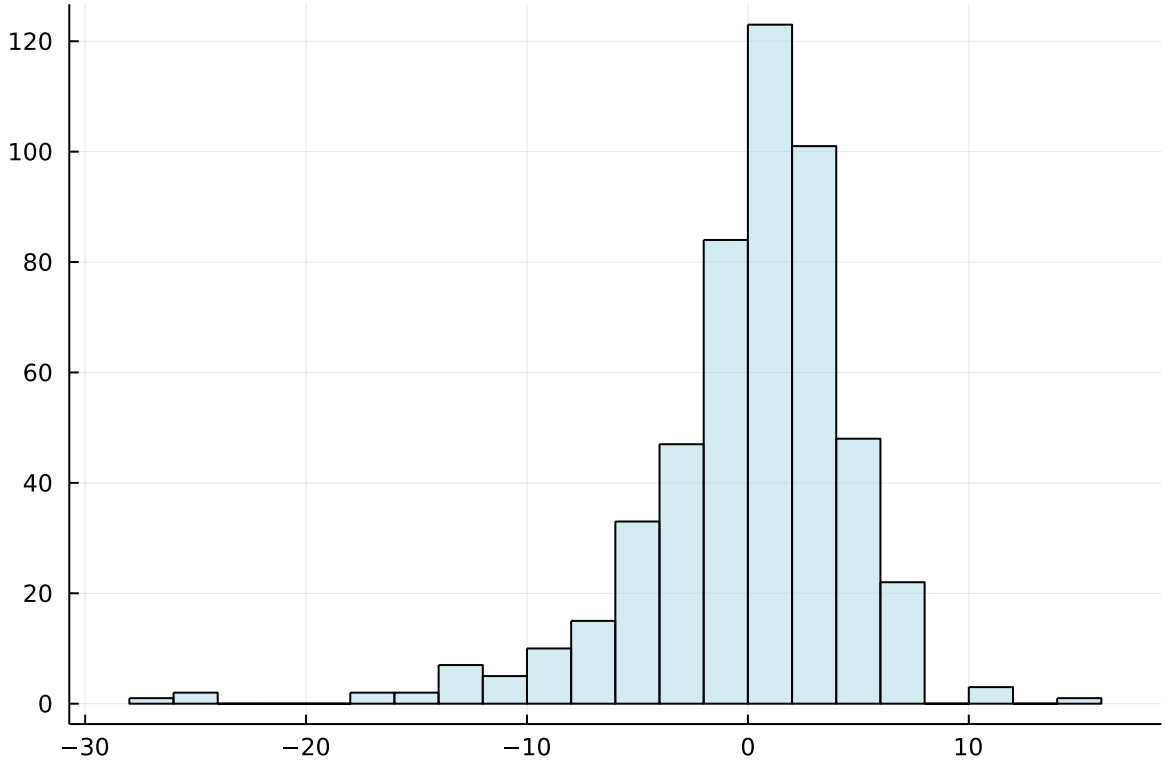
```
Crim    :  -0.108
Zn      :  0.0464
Indus   :  0.0206
Chas    :  2.69
NOx     :  -17.8
Rm      :  3.81
Age     :  0.000692
Dis     :  -1.48
Rad     :  0.306
Tax     :  -0.0123
PTRatio :  -0.953
Black   :  0.00931
LStat   :  -0.525
Intercept: 36.5
```

4.679

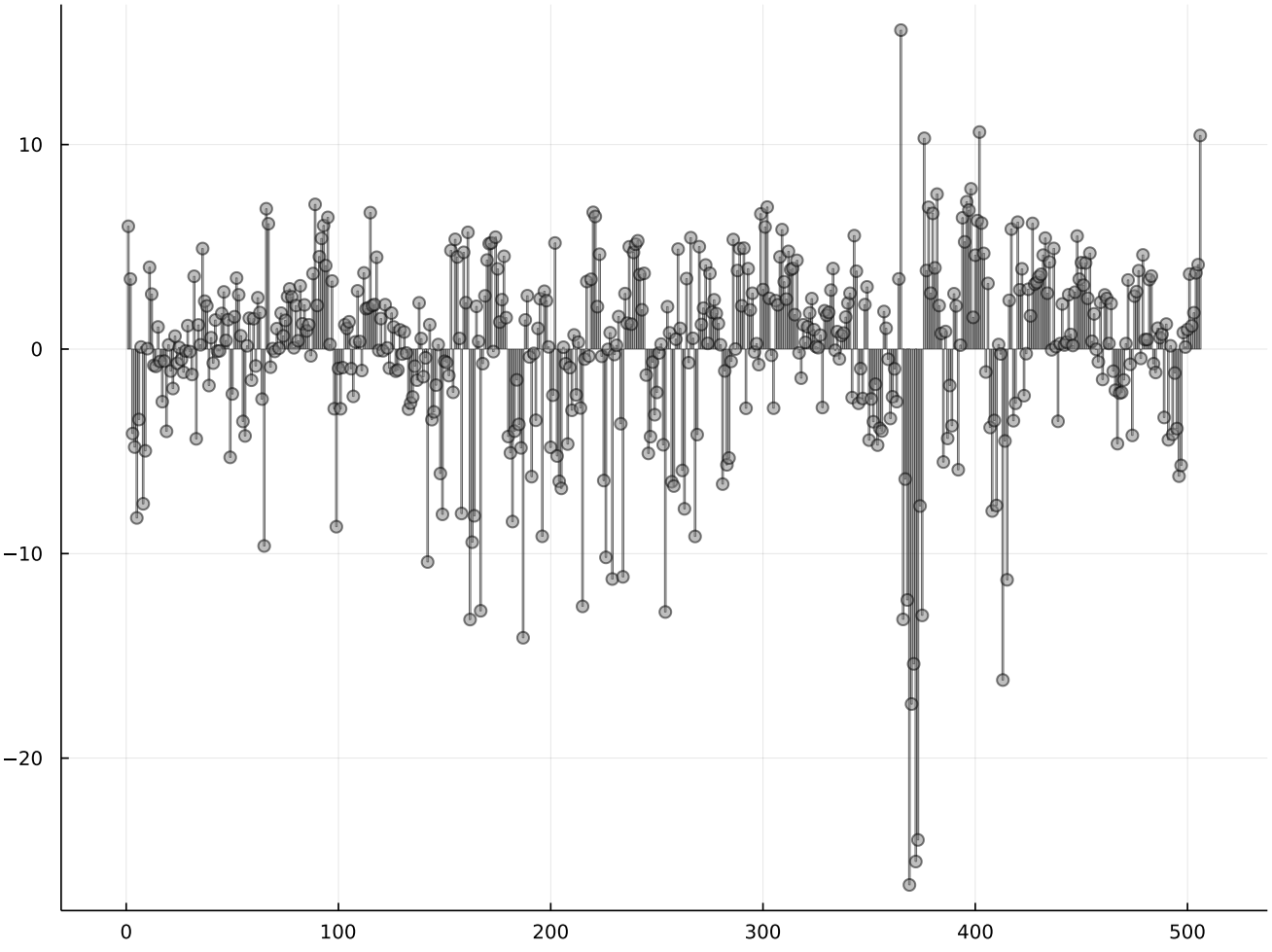
```
begin
    ŷ = MLJ.predict(mach, X)
    round(rms(ŷ, y), sigdigits=4)
end
```




```
begin
    n=size(boston)[1]
    range=1:n
    res = ŷ .- y      #    预测值和实际值逐一相减,得到每个数据点的残差集合
    plot(repeat((1:n)', 2),
         [zeros(1, n); res'], label = "", color = :grey, alpha = 0.5,size=(800,600))
    plot!(1:n, res, color = :grey, markershape = :circle,
          alpha = 0.5, label = "", linewidth = 0)
end
```




```
histogram(res,label=false,fc=:lightblue,fillalpha=0.5)  #残差频数统计柱形图
```




```
stem(res)
```

stem

stem

plot residual stem plot

```
stem(res::Array,color=:grey)
```

```
begin
    """
    stem

    plot  residual stem plot

    ```julia
    stem(res::Array,color=:grey)
    ```
    """
    function stem(res::Array,color=:grey)
        bar(res,fillcolor=color,width=0.5,label=false,alpha=0.5,size=(800,600))
        plot!(1:length(res), res, color = color, markershape = :circle,
              alpha = 0.5, label = "", linewidth = 0)

    end
end
```