

Data Mining and Machine Learning

Linear Regression

Dr. Edgar Acuña

<http://academic.uprm.edu/eacuna>

**UNIVERSIDAD DE PUERTO RICO
RECINTO UNIVERSITARIO DE MAYAGUEZ**

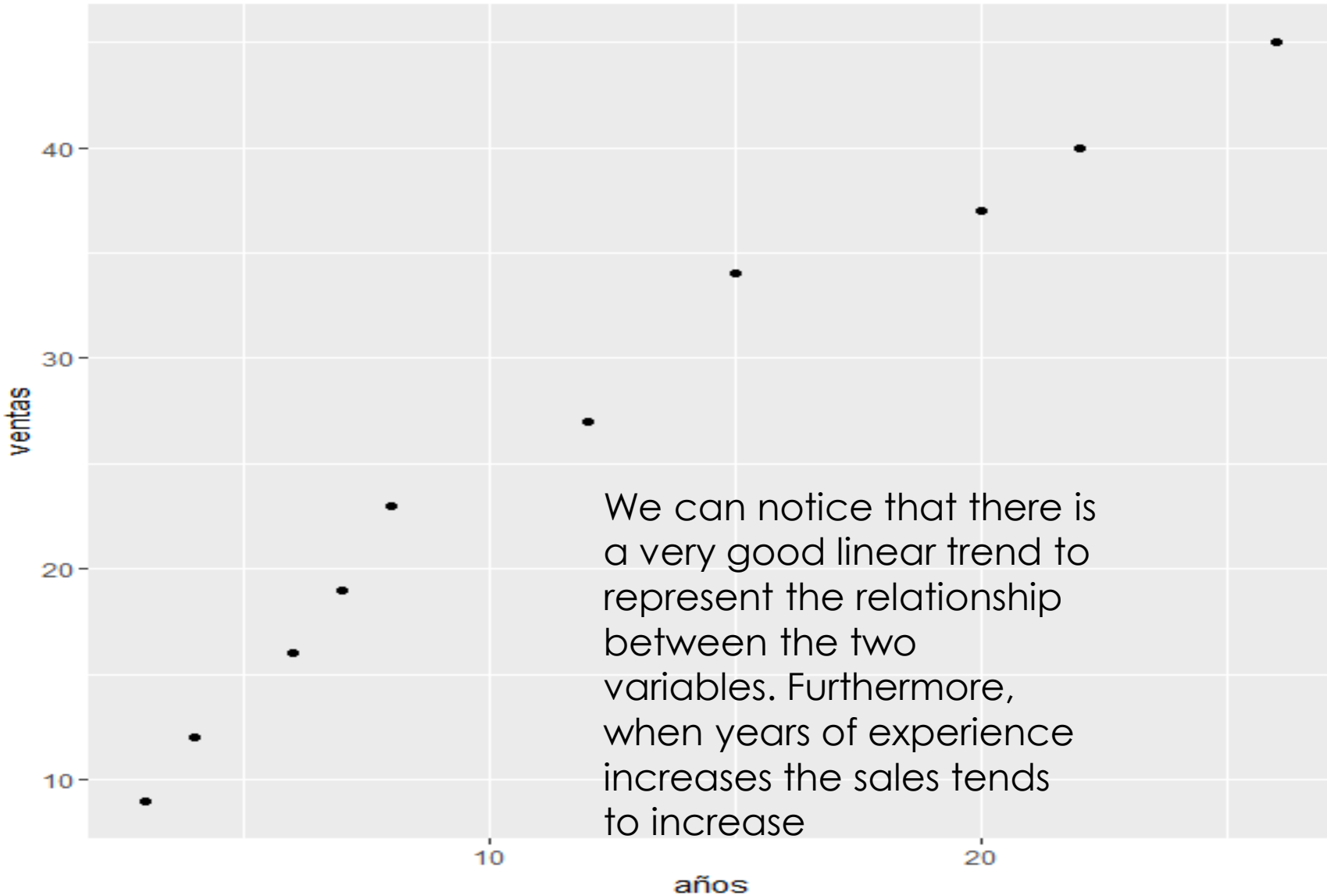
Example 1

A **car** dealer wants to predict car sales according to the year of experience of his salesmen. The following data represent years of experience of ten salesman (X) and the number of cars per year sold for them (Y).

X(años)	3	4	6	7	8	12	15	20	22	26
Y(ventas)	9	12	16	19	23	27	34	37	40	45

Solution:

First, we draw a scatterplot to check linearity.



2 The Correlation coefficient

Also, it is known as the Pearson correlation coefficient. It is represented by r and it measures the degree of linear association between two continuous attributes X and Y . It is computed as:

Se calcula por

$$r = \frac{S_{xy}}{\sqrt{S_{xx}S_{yy}}}$$

Donde:

$$S_{xx} = \sum_{i=1}^n x_i^2 - \frac{(\sum_{i=1}^n x_i)^2}{n}, \quad S_{yy} = \sum_{i=1}^n y_i^2 - \frac{(\sum_{i=1}^n y_i)^2}{n} \quad \text{y} \quad S_{xy} = \sum_{i=1}^n x_i y_i - \frac{(\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{n}$$

Both, S_{xx} and S_{yy} can not be negative, but S_{xy} could be either negative or positive.

- The correlation assumes values between -1 and 1.
- In most cases, a correlation greater than .75 or less than -.75 is considered good enough. On the other hand a correlation value between -.3 y .3 is considered low.
- If the correlation is positive then when X increases also Y tends to increase.
- If the correlation is negative the when X increases it is expected that Y decreases.

Example (cont)

Row	years	ventas	Sxx	Syy	Sxy	r
1	3	9	590.1	1385.6	889.4	0.983593
2	4	12				
3	6	16				
4	7	19				
5	8	23				
6	12	27				
7	15	34				
8	20	37				
9	22	40				
10	26	45				

En **Python**, the **correlation** coefficient can be found using several modules.

```
stats.pearsonr(years,ventas) #scipy
```

```
df.corr()["years"]["ventas"] #Pandas
```

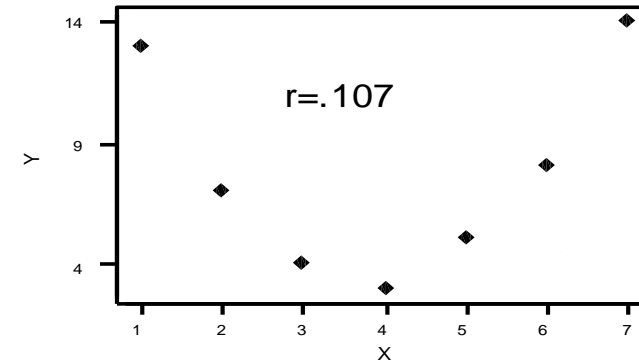
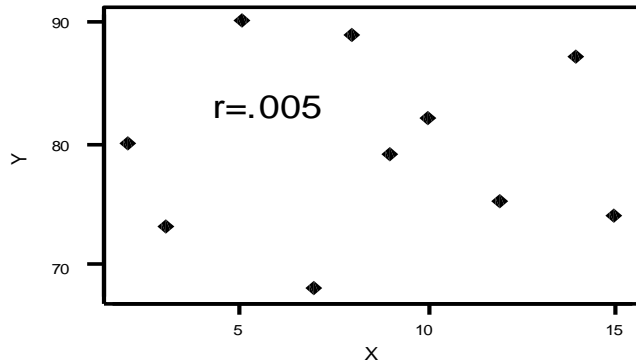
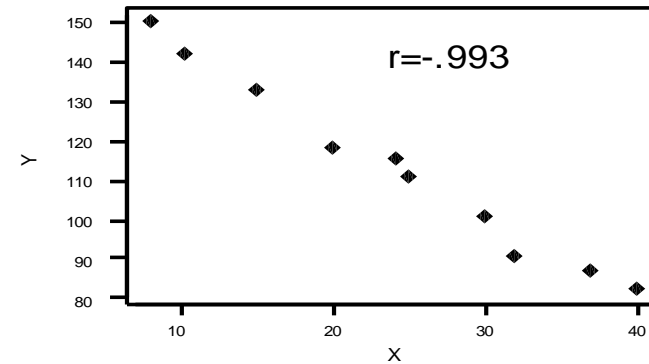
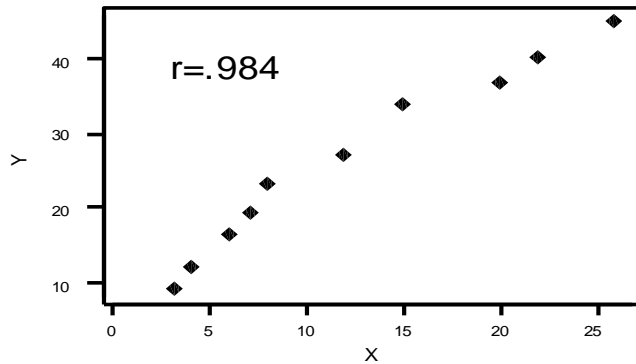
```
np.corcoeff(years,ventas) #numpy
```

```
.983593
```

Interpretation: *There is a very good linear relationship between years of experience of the salesman and the number of cars sold by him. Furthermore, to more year of experience of the salesman more cars he will sell. Years of experience can be used to predict sales through a linear equation.*

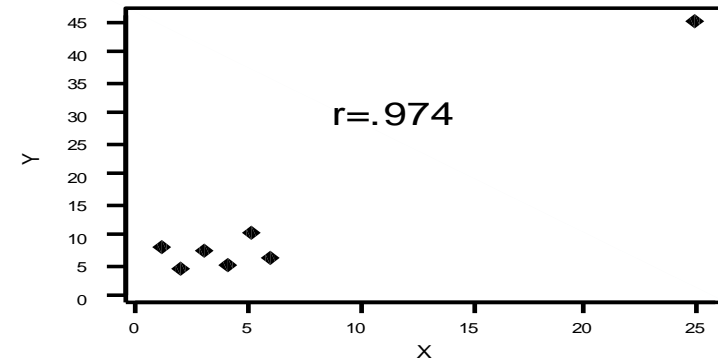
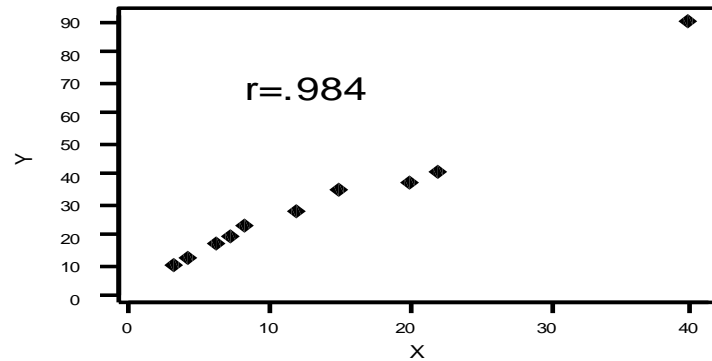
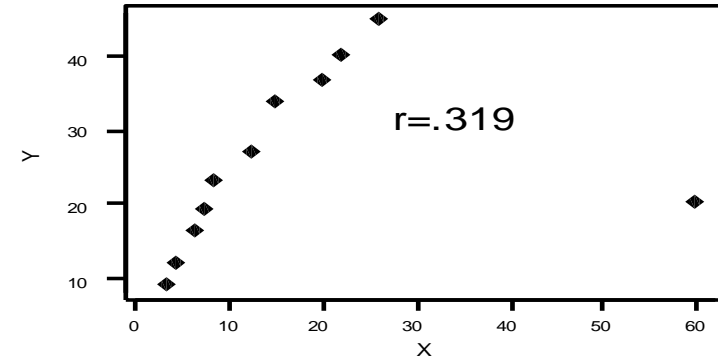
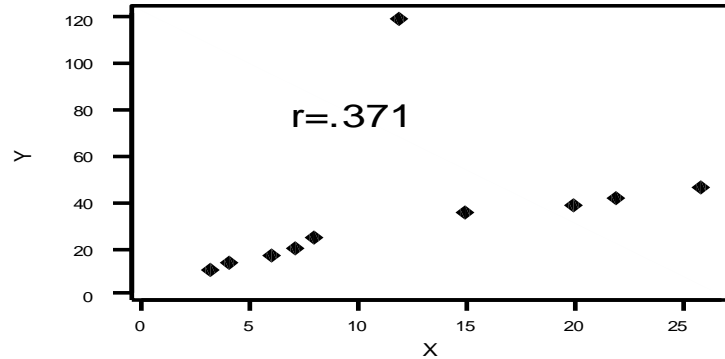
Correlation values for several scatterplots

Coeficiente de Correlacion para diversos plots



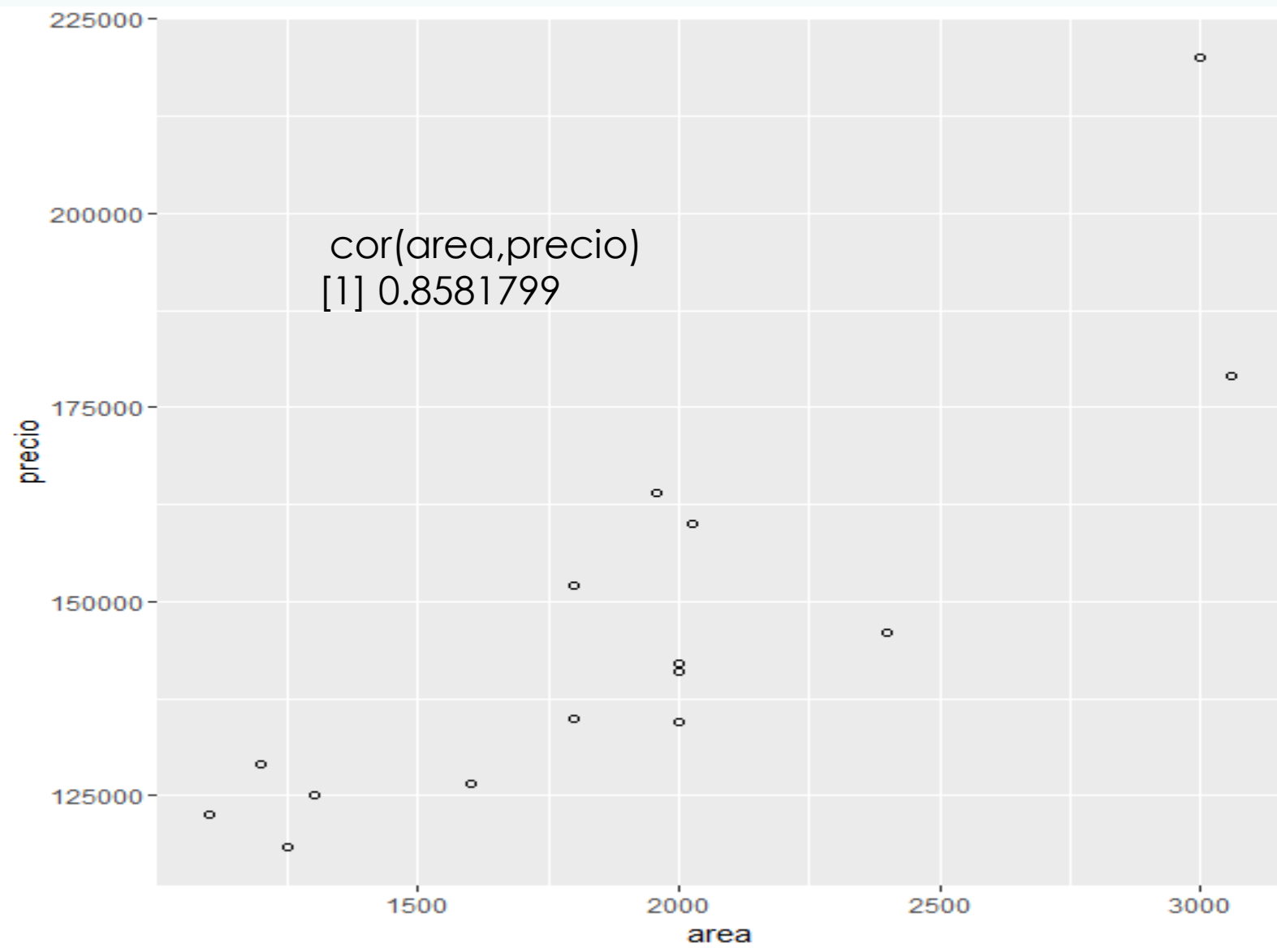
Effect of outliers in the correlation coefficient

Efecto de valores anormales en el valor de la correlacion



Example 2 <http://academic.uprm.edu/eacuna/casas.txt>

Casa	área(pies ²)	precio
1	3060	179000
2	1600	126500
3	2000	134500
4	1300	125000
5	2000	142000
6	1956	164000
7	2400	146000
8	1200	129000
9	1800	135000
10	1248	118500
11	2025	160000
12	1800	152000
13	1100	122500
14	3000	220000
15	2000	141000



Simple Linear Regression

The **simple linear regression model** is given by:

$$Y = \alpha + \beta X + \varepsilon$$

where, Y is the dependent or response variable,

X is the independent or predictor variable,

α is the y-intercept of the regression line,

β is the slope of the regression line, and

ε is a random error, which it is supposed to have mean 0 and constant variance σ^2 .

Estimation of the Regression Line

The linear regression model is estimated by solving the equation

$$\hat{Y} = \hat{\alpha} + \hat{\beta}X$$

The estimate $\hat{\alpha}$ of α and the estimate $\hat{\beta}$ of β are found using the least squares technique, which is based on minimizing the error sum of squares.

$$Q(\alpha, \beta) = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2$$

and it is obtained

$$\hat{\beta} = \frac{s_{xy}}{s_{xx}} \qquad \hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$$

Estimated linear regression for example 1

```
result = sm.ols(formula="ventas ~ years", data=df).fit()
print result.params
print result.summary()
```

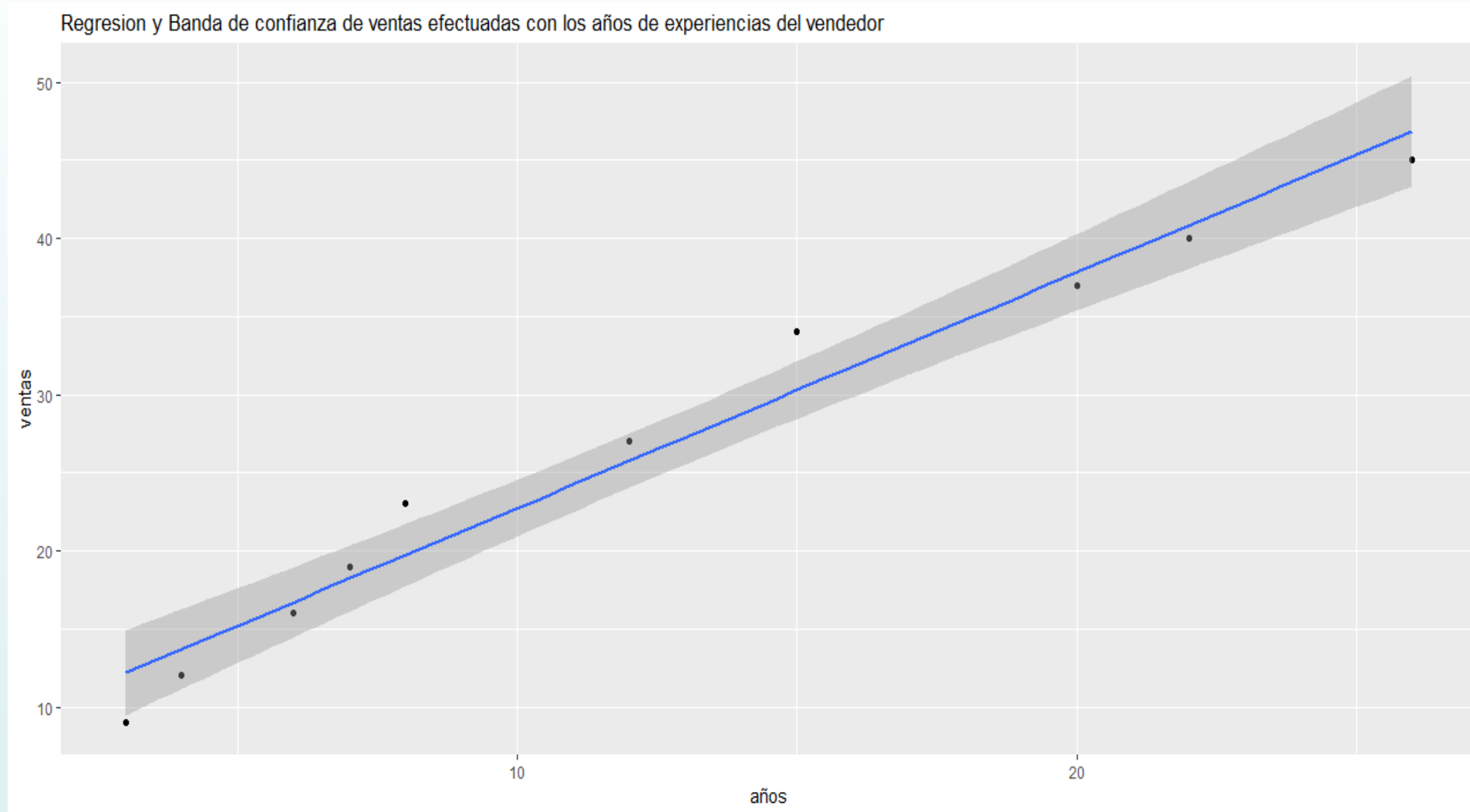
Intercept 7.661413

years 1.507202

OLS Regression Results

```
=====
=== Dep. Variable: ventas R-squared: 0.967
Model: OLS                      Adj. R-squared: 0.963
Method: Least Squares          F-statistic: 237.8
Date: Wed, 24 Oct 2018         Prob (F-statistic): 3.11e-07
Time: 21:07:51                 Log-Likelihood: -21.720
No. Observations: 10           AIC: 47.44
Df Residuals: 8                 BIC: 48.05
=====
```

```
=====
              coef    std err   t      P>|t| [0.025 0.975] -----
Intercept 7.6614  1.417   5.405  0.001  4.393 10.930
years     1.5072  0.098  15.421  0.000  1.282 1.733
=====
```



This plot can be obtained with plotnine or seaborn

Estimated linear regression for example 2

```
result = sm.ols(formula="precio ~ area", data=df).fit()
```

```
print result.params
```

```
print result.summary()
```

```
Intercept 73167.748381
```

```
area 38.523071
```

OLS Regression Results

=====

Dep. Variable: precio R-squared: 0.736

Model: OLS Adj. R-squared: 0.716

Method: Least Squares F-statistic: 36.33

Date: Wed, 24 Oct 2018 Prob (F-statistic): 4.25e-05

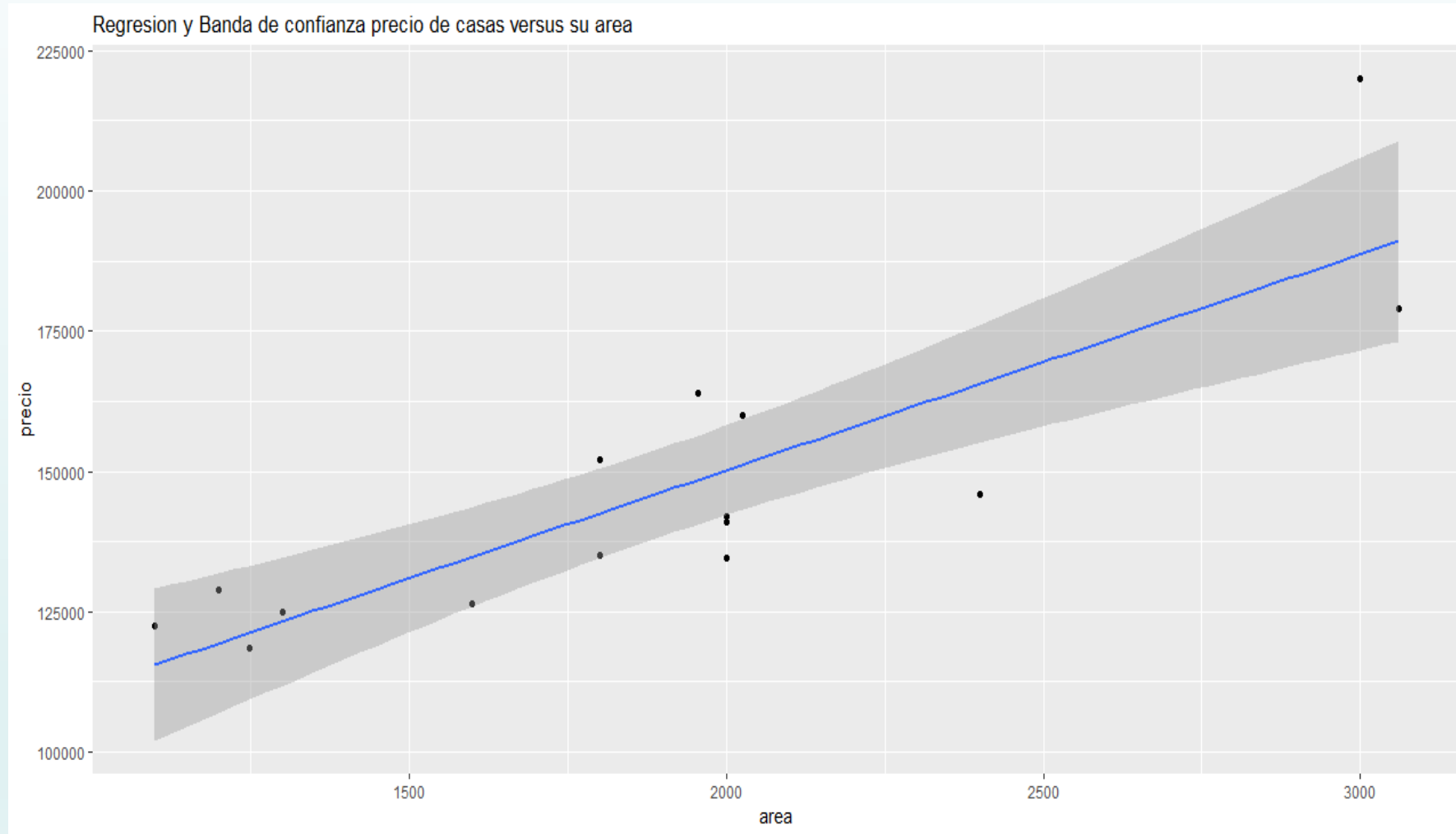
Time: 21:09:58 Log-Likelihood: -163.54 No.

Observations: 15 AIC: 331.1

Df Residuals: 13 BIC: 332.5

=====

	coef	std err	t	P> t	[0.025	0.975]	---
Intercept	7.317e+04	1.27e+04	5.773	0.000	4.58e+04	1.01e+05	
area	38.5231	6.391	6.028	0.000	24.716	52.330	



Interpretation of the regression coefficients

- **Interpretation of intercept** $\hat{\alpha}$

It means the average value of the response variable Y when X is zero. If one knows for sure that the predictor X can not assume a 0 value then it does not make sense to interpret the intercept.

In example 2, $\hat{\alpha} = 73,168$ would mean that the average price of houses without is 73,158, something really unreasonable.

- **Interpretation of slope** : $\hat{\beta}$

It means the expected change in the response variable Y when the predictor X increases in one unit.

In example 2, ~~38.5~~ 38.5 means that for each additional feet square of área the price of the house it is expected to increase in 38.5 dollars.

Inference in lineal Regression

- **Inference about the regression coefficients**

The most frequent hypothesis testing are, $H_0: \alpha = 0$ versus $H_a: \alpha \neq 0$ and $H_0: \beta = 0$ versus $H_a: \beta \neq 0$.

The statistical test for the slope is given by:

$$t = \frac{\hat{\beta}}{s.e(\hat{\beta})} = \frac{\hat{\beta}}{\frac{s}{\sqrt{S_{xx}}}} \quad \text{y} \quad s = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n-2}}$$

This test has a t distribution with $n-2$ degrees of freedom.

The module **statsmodels** gives the value of the statistical t-test and its respective p-value. A p-value near to 0, say less than 0.05, leads us to reject the null hypothesis.

If the null hypothesis is rejected it means that somehow the variable X is important to predict the value of Y using a regression line. But, if the null hypothesis is not rejected then one concludes that the variable X can not be used to predict Y using a regression line.

In the example 2 the value of the statistical t test for the slope is 6.028 and its P-value = .0000 then the conclusion is to reject the null hypothesis. There is sufficient statistical evidence to conclude that the variable area of the house can be used to predict its price.

The Determination Coefficient

It is measure of goodness of fit for the estimated regression model.

$$R^2 = \frac{SSR}{SST}$$

where,

SSR represents the sum of squares due to regression

SST represents the total sum of squares.

Notice that the Determination coefficient is the square of the correlation coefficient.

The Determination coefficient varies between 0 and 1, although is very common to express it in percentage. A R^2 greater than 70 % indicates a good linear association between the two variables, and the X variable can be used to predict Y. Also, R^2 indicates which percentage of the total variation of the response variable Y is explained by its linear relationship with X.

In the example 2, $R^2=73.6$, this means that 73.6% of the variability of the home prices is explained by its linear relationship with the its área. The variable área may be used to predict the predict the Price of a house.

Multiple Linear Regression

The multiple linear regression model with p predictor variables X_1, \dots, X_p , is given by

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip} + e_i$$

The constants $\beta_0, \beta_1, \beta_2, \dots, \beta_p$, are called regression coefficients and they are estimated by the least squares method using a training dataset of n observations of the form:

$y_i, x_{i1}, x_{i2}, \dots, x_{ip}$, for $i = 1, \dots, n$. The term ε_i is a random variable with mean 0 and variance σ^2 .

Among the Python' modules for linear regression are: Statsmodels, el submodule linear_model from sklearn includes Linear regression and tensorflow includes a LinearRegression function in its sub-module estimator. Also, Tensorflow can compute linear regression using the Gradient Descent method to minimize the Sum of Squares Error.

The multiple linear regression model

In matrix form : $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e}$

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}$$

Interpretación del coeficiente de regresión estimado β_j

The estimated of the poblacional regression coefficient, b_j , for $j = 1, \dots, p$ is represented by β_j .

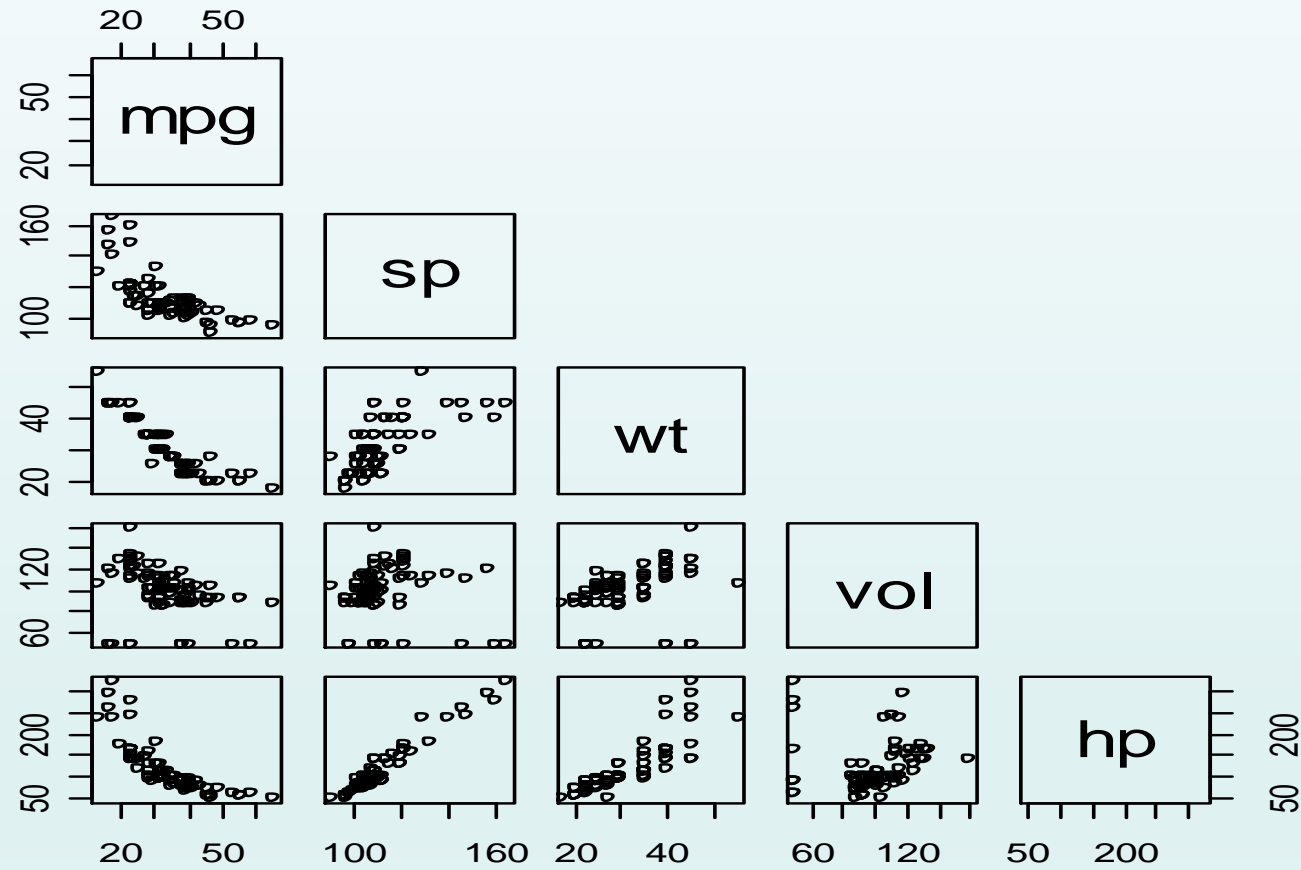
This estimated indicates the expected change in the response variable Y when the predictor variable X_j changes in one additional unit assuming that the other predictors remain constant.

Example

Consider the **millaje** dataset, where the response variable is Y = (MPG) miles per gallon of a car, and the predictor variables are:
 X_1 =(VOL): volume,
 X_2 =(HP): horsepower,
 X_3 = (SP) :maximum speed and,
 X_4 =(WT): weight.

The data is available at academic.uprm.edu/eacuna/millaje.txt.

Matrix plot



Least Squares estimation of the parameter vector β

The error sum of squares must be minimized

$$Q(\beta) = \sum_{i=1}^n e_i^2 = \mathbf{e}'\mathbf{e} = (\mathbf{Y} - \mathbf{X}\beta)'(\mathbf{Y} - \mathbf{X}\beta)$$

Derivating Q with respect to β and equating to zero, the following system of normal equations is obtained

$$\mathbf{X}'\mathbf{X}\beta = \mathbf{X}'\mathbf{Y}$$

Solving for β the following vector of estimated coefficients is obtained

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

Least Squares estimation of the parameter vector β (2)

```
regall=smf.ols(formula='mpg~sp+wt+vol+hp',data=data1).fit()
print regall.summary()
```

```
=====
Dep. Variable: mpg          R-squared: 0.873
Model: OLS                 Adj. R-squared: 0.867
Method: Least Squares      F-statistic: 132.7
Date: Mon, 19 Mar 2018     Prob (F-statistic): 9.98e-34
Time: 12:57:26             Log-Likelihood: -220.00
No.Observations: 82        AIC: 450.0
Df Residuals: 77          BIC: 462.0
=====
```

```
=====
              coef    std err   t      P>|t|  [0.025 0.975]
Intercept 192.4378   23.532   8.178   0.000   145.580 239.295
Sp        -1.2948    0.245  -5.290   0.000   -1.782  -0.807
wt        -1.8598    0.213  -8.717   0.000   -2.285  -1.435
vol       -0.0156    0.023  -0.685   0.495   -0.061   0.030
hp         0.3922    0.081   4.818   0.000    0.230   0.554
=====
```

Interpretation of the estimated vector of parameters β

- The estimated regression model for the mpg data is the following:
$$\text{MPG} = 192 - 1.29 \text{ SP} - 1.85 \text{ WT} - 0.01 \text{ VOL} + 0.39 \text{ HP}$$
- Interpretation of the coefficients:
- $B_1 = -1.29$ means that if the maximum speed increases in one m/h then it is expected that the mpg decreases in 1.29, assuming that the other variables remain constant.
- $B_2 = -1.85$ means that if the car's weight increases in one unit (100 lb) then it is expected that the mpg decreases in 1.85, assuming that the other variables remain constant.
- $B_3 = -.01$ means that if the interior volumen of the car increases in one feet cubic then it is expected that the mpg decreases in 1.85 assuming that the other variables remain constant.
- $B_4 = 0.39$ means that if the horsepower increases in one unit then it is expected that the mpg increases in 0.39, assuming that the other variables remain constant.

Prediction

```
#Predicting the mpg for a car with sp=100, wt=20, vol=100 and hp=50
pred=regall.predict(pd.DataFrame({'sp': [100], 'wt':[20], 'vol': [100], 'hp':
    [50]}))
print "The predicted mpg is:",pred[0]
The predicted mpg is: 43.8059456465
```

Inference in multiple linear Regression

It includes:

- Hypothesis testing and confidence intervals about the coefficients of the population regression model.
- Confidence Interval for predictions made with the model

Hypothesis testing about regression coefficients

- $H_0: \beta_i = 0$ ($i=1,2,\dots,p$),
- $H_a: \beta_i \neq 0$;

The statistical test is a t test:

$$t = \frac{\hat{\beta}_i}{se(\hat{\beta}_i)} = \frac{\hat{\beta}_i}{s\sqrt{C_{ii}}} \quad \text{which has a t distribution with } (n-p-1) \text{ df.}$$

where, C_{ii} is the i -th element in the diagonal of $(\mathbf{X}'\mathbf{X})^{-1}$.

Statmodels gives the “P-value” for the t-test. When the P-value is < 0.05 H_0 is rejected.

Hypotesis for millaje

coef	std err	t	P> t	[0.025	0.975]
Intercept	192.4378	23.532	8.178	0.000	145.580 239.295
Sp	-1.2948	0.245	-5.290	0.000	-1.782 -0.807
wt	-1.8598	0.213	-8.717	0.000	-2.285 -1.435
vol	-0.0156	0.023	-0.685	0.495	-0.061 0.030
hp	0.3922	0.081	4.818	0.000	0.230 0.554

According to the above table all the predictors but vol are significant since their p-value is 0.000 less than .05

Feature selection in Multiple Linear Regression

- Backward Elimination
- Forward Selection
- Stepwise

Backward Elimination method

At the initial step all the predictors are included in the model, and in each step the variable less important is eliminated from the model. The importance of the variable is determined according to several criteria. For instance, the variable which p-value is the highest is the less important one. Also, one can look at the value of the t-tests.

A variable which is eliminated from the model can not be considered anymore.

The process ends when all the variables have p-values less than .05. This is the same that to have all the t-tests greater than 2 in absolute value.

Statmodels uses instead of the t-test the Akaike's Information criterion (AIC).

The AIC criterion

Akaike's Information Criterion (Akaike, 1973) uses concepts of information theory. It is based on the minimization of the Kullback-Leibler distance between the distribution of the response variable Y under the reduced model and under the full model.

The AIC is calculated as:

$$\text{AIC} = n \cdot \log[\text{SSE}_p/n] + 2p \quad (1)$$

where n , is the number of observations, p is the number of parameters in the model, and SSE is the error sum of squares of the regression model.

There are other versions of the formula (1).

The best model is the one with the lowest AIC.

Example. The dataset **grasa** contains 13 variables to predict the percentage of body fat

Columna	Nombre
v1	grasa (% de grasa)
v2	edad (en años)
v3	peso (en libras)
v4	altura (en pulgadas)
v5	cuello (en cms)
v6	pecho (en cms)
v7	abdomen (en cms)
v8	cadera (en cms)
v9	muslo (en cms)
v10	rodilla (en cms)
v11	tobillo (en cms)
v12	biceps (en cms)
v13	antebrazo (en cms)
v14	muñeca (en cms)

The measurements were taken in 252 subjects

The data is available at <http://academic.uprm.edu/eacuna/grasa.txt>

```
model=backward_elimination(data2,"grasa")
print model
AIC including all the features
1464.50237446
Feature considered for elimination in this step: rodilla
AIC 1462.50659712
Feature considered for elimination in this step: pecho
AIC= 1460.57130604
Feature considered for elimination in this step: altura
AIC= 1459.06626108
Feature considered for elimination in this step: tobillo
AIC= 1457.82217075
Feature considered for elimination in this step: biceps
AIC= 1456.99638198
Feature considered for elimination in this step: cadera
best AIC: 1456.99638198
Best Features: ['edad', 'peso', 'cuello', 'abdomen', 'cadera', 'muslo', 'antebrazo',
'muneca'] grasa ~ edad + peso + cuello + abdomen + cadera + muslo + antebrazo +
muneca + 1
```

```

#Aplicandolo Backward Elimination de MLxtend a grasa
a1=SFS(lr, k_features=5, forward=False, floating=False, verbose=0, cv=None)
a1.fit(X1,y1)
a1.subsets_
{13: {'feature_idx': (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12),
{13: {'feature_idx': (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12),
'cv_scores': array([0.74904997]),
'avg_score': 0.749049969148275,
'feature_names': ('0', .....
12: {'feature_idx': (0, 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12),
'cv_scores': array([0.74904576]),
'avg_score': 0.7490457640449057,
'feature_names': ('0', .....
11: {'feature_idx': (0, 1, 2, 3, 5, 6, 7, 9, 10, 11, 12),
'cv_scores': array([0.74898132]),
'avg_score': 0.7489813153923306,
'feature_names': ('0', '1', '2', '3', '5', '6', '7', '9', '10', '11', '12')},
10: {'feature_idx': (0, 1, 3, 5, 6, 7, 9, 10, 11, 12),
'cv_scores': array([0.7484878]),
'avg_score': 0.7484878032606919,
'feature_names': ('0', '1', '3', '5', '6', '7', '9', '10', '11', '12')},
9: {'feature_idx': (0, 1, 3, 5, 6, 7, 10, 11, 12),
'cv_scores': array([0.74773222]),
'avg_score': 0.747732224159076,
'feature_names': ('0', '1', '3', '5', '6', '7', '10', '11', '12')},

```

```
8: {'feature_idx': (0, 1, 3, 5, 6, 7, 11, 12),  
    'cv_scores': array([0.74655402]),  
    'avg_score': 0.7465540223833473,  
    'feature_names': ('0', '1', '3', '5', '6', '7', '11', '12')},  
7: {'feature_idx': (0, 1, 3, 5, 7, 11, 12),  
    'cv_scores': array([0.7444763]),  
    'avg_score': 0.7444762951356895,  
    'feature_names': ('0', '1', '3', '5', '7', '11', '12')},  
6: {'feature_idx': (0, 1, 5, 7, 11, 12),  
    'cv_scores': array([0.74096806]),  
    'avg_score': 0.7409680552099627,  
    'feature_names': ('0', '1', '5', '7', '11', '12')},  
5: {'feature_idx': (0, 1, 5, 11, 12),  
    'cv_scores': array([0.73719345]),  
    'avg_score': 0.7371934537698608,  
    'feature_names': ('0', '1', '5', '11', '12')}}}
```

Interpretation: The best model is:

```
backbest=smf.ols(formula='grasa ~ edad + peso + cuello + abdomen + cadera +  
muslo + antebrazo + muneca',data=data2).fit()
```

```
print backbest.params
```

```
Intercept -22.656373 edad 0.065780 peso -0.089853 cuello -0.466558 abdomen  
0.944815 cadera -0.195435 muslo 0.302392 antebrazo 0.515721 muneca -  
1.536652
```

```
print 'R2= ', backbest.rsquared
```

```
R2= 0.746554022383
```

The best model has a R^2 equals to 74.66, whereas the full model using the 13 predictors has a $R^2 = 74.90\%$. A 0.24% of prediction power has been lost, however the best model has only 8 predictors. Therefore the second model is more convenient.

Forward Selection

At the initial step, the single best variable is the one having the highest correlation with the response variable.

At the second step, it is included the variable that along with the first one improves a metric such as the R-squared or the AIC. The process continues until the metric can not be improved.

Notice that one variable that is elected can not be excluded from the model in a later step.

Example (cont). En el primer paso se halla la regresión simple con la variable predictora más altamente correlacionada con la variable de respuesta. En este caso, es *abdomen* que tiene correlación 0.803 con *grasa*.

La segunda variable que entra al modelo es *peso* porque es aquella con el valor de t más grande en valor absoluto entre las doce variables que aún no estaban incluidas.

```
model=forward_aic_selection(data2,"grasa")  
print model.model.formula  
print model.aic
```

```
1515.79033747 abdomen  
1471.18477769 peso  
1465.04121338 muneca  
1460.219691 antebrazo  
1459.44186161 cuello  
1458.80625316 edad  
1457.05383331 muslo  
1456.99638198 cadera  
1457.82217075 biceps  
grasa ~ abdomen + peso + muneca + antebrazo + cuello + edad + muslo + cadera + 1  
1456.99638198
```

Stepwise

Es una modificación del método “Forward”, donde una variable que ha sido incluida en el modelo en un paso previo puede ser eliminada posteriormente.

En cada paso se cotejan si todas las variables que están en el modelo deben permanecer allí. La mayoría de las veces, pero no siempre, los tres métodos dan el mismo resultado para el mejor modelo de regresión.

```

#Aplicandolo a grasa
a1=SFS(lr, k_features=5, floating=True, verbose=0, cv=None)
a1.fit(X1,y1)
a1.subsets_
{1: {'feature_idx': (5,),
    'cv_scores': array([0.66167208]),
    'avg_score': 0.6616720819241175,
    'feature_names': ('5',)},
 2: {'feature_idx': (1, 5),
    'cv_scores': array([0.7187981]),
    'avg_score': 0.7187981048369947,
    'feature_names': ('1', '5')},
 3: {'feature_idx': (1, 5, 12),
    'cv_scores': array([0.72774009]),
    'avg_score': 0.7277400882138524,
    'feature_names': ('1', '5', '12')},
 4: {'feature_idx': (1, 5, 11, 12),
    'cv_scores': array([0.73501118]),
    'avg_score': 0.7350111804231527,
    'feature_names': ('1', '5', '11', '12')},
 5: {'feature_idx': (1, 3, 5, 11, 12),
    'cv_scores': array([0.73791615]),
    'avg_score': 0.7379161469620583,
    'feature_names': ('1', '3', '5', '11', '12')}}

```