STAT1005 Foundations of Data Science

Lecture (8): Regression & Prediction

Yuanhua Huang (黃淵華)

Office: Rm 1-05E, 1/F, JCBIR, 5 Sassoon Road (Medical Campus)
Q&A contact hours: Wed 3-5pm

Email: yuanhua@hku.hk | Web: https://web.hku.hk/~yuanhua

Nov 1, 2021

Department of 統計及精算學系
Statistics & Actuarial Science
THE UNIVERSITY OF HONG KONG

HKU Med
LKS Faculty of Medicine
School of Biomedical Sciences
香港大學生物醫學學院

Objectives today

1. Simple linear regression (One explanatory variable)
   1. Concept and mathematics behind the model
   2. Implement linear regression from scratch
2. Multiple linear regression (Multiple explanatory variable)
   1. The least square estimators
   2. Towards a wise selection of input variables
3. Exploring non-linear relationship
   1. Data transformation

- Wiki: https://en.wikipedia.org/wiki/Linear_regression
- Scikit learn: https://scikit-learn.org/stable/modules/linear_model.html
- statsmodels: https://www.statsmodels.org/stable/regression.html
- Seaborn regplot: https://seaborn.pydata.org/generated/seaborn.regplot.html
- Notebooks: https://github.com/huangyh09/foundation-data-science/

Forward | Machine learning (big picture for future study)

Machine learning categories

- Supervised learning: relationship between input and output f(X) → y
  - y is continuous: regression (this lecture)
  - y is categorical: classification (next lecture)
- Unsupervised learning: understand the underlying patterns behind X, *P*(X)
  - Discrete groups: clustering
  - Continuous factors: dimension reduction or latent factor models
- Reinforcement learning:
  - How intelligent agents ought to take actions in an environment in order to maximize the notion of cumulative reward
  - It has reward but not immediately together with input at the same time

Forward | Linear regression models (today)

Linear regression (linear model)

- Predictive model: basic technique that uses historical data to predict an output variable
- Linear relationship: between input and output (which is actually common)
- Easy interpretation: makes it a popular tool as can be explained using plain English and can be visualised using simple line graphs.
- Handy build-in function: Python has several ready-made package with detailed documentations, making predictive analytics easy!

- It is however important to understand what the code is doing behind the scenes
  - How does the model work?
  - What do the results mean?
  - What are the assumptions?

Acknowledgement: these notes are based on last year’s slides from Prof. Jeff Yao

Linear regression | Some terminology

- A predictive model is a mathematical equation consisting of input variables yielding an output when values of the input variables are provided.

- For example, let’s assume that the price *P* of a house is linearly dependent upon its size *S*, amenities *A*, and availability of transport *T*. The equation will look like this:

$$P = a_0 + a_1 * S + a_2 * A + a_3 * T, (1)$$

- Equation (1) is called a model, in fact a multiple linear regression model.

Simple linear regression: one explanatory variable;

Multiple linear regression: multiple explanatory variables.

Linear regression | Some terminology

$$P = a_0 + a_1 * S + a_2 * A + a_3 * T, \quad (1)$$

- *P*, *S*, *A*, and *T* are variables. They can be numerical (continuous), binary or categorical.
- The variable *P* is the output variable, also known as the predicted output (dependent variable).
- *S*, *A*, and *T* are input variables, also known as predictors (covariates, independent variables, explanatory variables or features).
- a_0, a_1, a_2, a_3 are called variable coefficients, weights or, more commonly, parameters of the model. These parameters are unknown.

Our job is to estimate the parameters using historical input and output data.

Simple linear regression | Understand the maths behind

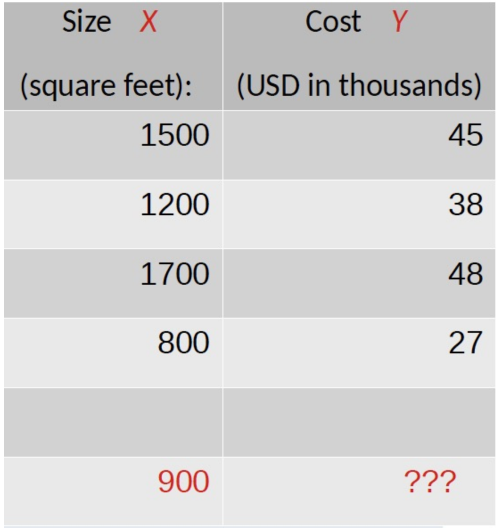
- Consider a hypothetical dataset containing information about the costs of several houses and their sizes (in square feet);

- Aim: model the output variable $y = Cost$ in function of the input variable $x = Size$

- To estimate y using linear regression, we assume that y is a linear function of x : $\hat{y} = \alpha + \beta * x$

  where $\hat{y}$ is the estimated or predicted value of y based on our linear equation.

In our example, $Cost = \alpha + \beta * Size$



Size X (square feet):	Cost Y (USD in thousands)
1500	45
1200	38
1700	48
800	27
900	???

Simple linear regression | Understand the maths behind

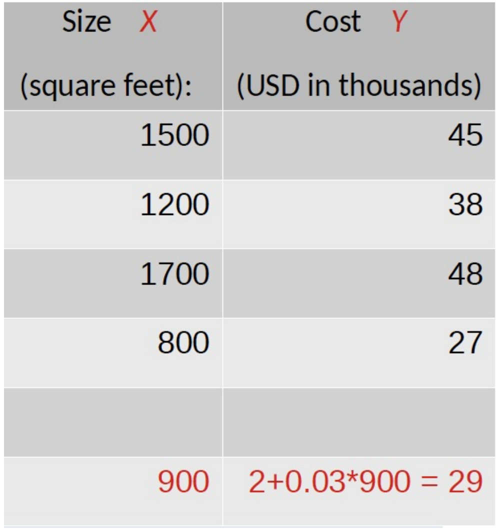
y is a linear function of x : $\hat{y} = \alpha + \beta * x$

- The purpose of linear regression is to find statistically significant values of α and β that minimize the difference between y and $\hat{y}$.

- If we can determine the optimum values of these two parameters α and β, then we will have an equation that we can use to predict the values of y, given the value of x.

- E.g., if we find α = 2 and β = 0.03, then the equation will be $\hat{y} = 2 + 0.03 * x$;

Using this equation, we can find the cost of a home of any size.



Size X (square feet):	Cost Y (USD in thousands)
1500	45
1200	38
1700	48
800	27
900	2+0.03*900 = 29

For a 900 ft^2 house:

Cost = α + β * Size

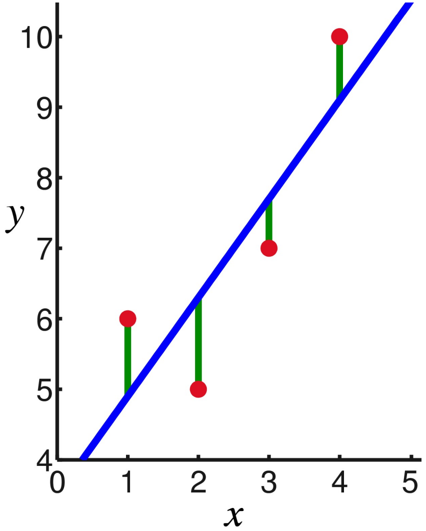
= 2 + 0.03 * 900

= 29 (10^3 USD)

Simple linear regression | Estimating the parameters

How to obtain a good guess of α and β

- Using the method of Least Squares (Laplace, Gauss)
- Green lines show the difference between actual values y and estimate values $\hat{y}$



- The objective of the least squares method is to minimize the *sum of the squared error (difference), SSE,* between y and $\hat{y}$. This can be written as (with n data points):

$$SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - (\alpha + \beta x_i))^2$$

- Using calculus, we can show that the values of the optimal parameters α and β are as follows:

$$\beta = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}; \quad \alpha = \bar{y} - \beta\, \bar{x}$$

where $\bar{x}$ is the mean of x values, and $\bar{y}$ the mean of y values

Details of the derivation (for those who are curious!)

❶ Let $S(\alpha, \beta)$ be the sum of the squares in (1). Take partial derivatives of S w.r.t. the parameters α and β and equate to 0:

$$\frac{\partial S(\hat{\alpha}, \hat{\beta})}{\partial \hat{\alpha}} = -2 \sum_{i=1}^{n} (y_i - \hat{\alpha} - \hat{\beta} * x_i) = 0 \qquad \text{(I)}$$

$$\frac{\partial S(\hat{\alpha}, \hat{\beta})}{\partial \hat{\beta}} = -2 \sum_{i=1}^{n} (y_i - \hat{\alpha} - \hat{\beta} * x_i) x_i = 0 \qquad \text{(II)}$$

❷ Solve these equations to get the values of the parameters α and β:

From (I):

$$\sum_{i=1}^{n} (y_i - \hat{\alpha} - \hat{\beta} * x_i) = 0$$

$$\Longrightarrow \sum_{i=1}^{n} y_i - n\hat{\alpha} - \sum_{i=1}^{n} \hat{\beta} x_i = 0$$

$$\Longrightarrow \quad n\bar{y} - n\hat{\alpha} - n\hat{\beta}\bar{x} = 0$$

$$\Longrightarrow \quad \bar{y} = \hat{\alpha} + \hat{\beta}\bar{x}.$$

From (II):

$$\sum_{i=1}^{n} (y_i - \hat{\alpha} - \hat{\beta} x_i) x_i = 0$$

$$\Rightarrow \sum_{i=1}^{n} y_i x_i - \hat{\alpha} x_i - \hat{\beta} x_i^2 = 0$$

$$\Rightarrow \sum_{i=1}^{n} y_i x_i - (\bar{y} - \hat{\beta}\bar{x}) x_i - \hat{\beta} x_i^2 = 0$$

$$\Rightarrow \sum_{i=1}^{n} y_i x_i - \bar{y} x_i + \hat{\beta}\bar{x} x_i - \hat{\beta} x_i^2 = 0$$

$$\Rightarrow \sum_{i=1}^{n} (y_i - \bar{y}) x_i + \hat{\beta}(\bar{x} - x_i) x_i = 0$$

$$\Rightarrow \sum_{i=1}^{n} (y_i - \bar{y}) x_i = \hat{\beta} \sum_{i=1}^{n} (x_i - \bar{x}) x_i$$

$$\Rightarrow \hat{\beta} = \frac{\sum_{i=1}^{n} (y_i - y) x_i}{\sum_{i=1}^{n} (x_i - x) x_i}$$

$$\Rightarrow \hat{\beta} = \frac{\sum_{i=1}^{n} (y_i - y)(x_i - x)}{\sum_{i=1}^{n} (x_i - x)(x_i - x)}$$

Because of

$$\sum_{i=1}^{n} (y_i - \bar{y}) \bar{x} = 0; \ \sum_{i=1}^{n} (x_i - \bar{x}) \bar{x} = 0$$

Simple linear regression | Diagnostic statistics

SSE: sum of squares error (difference; SSD)

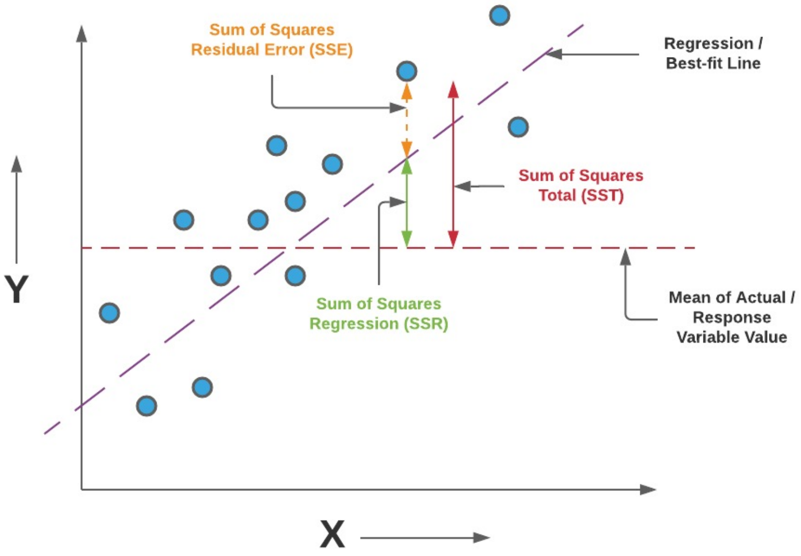
$$SSD = SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} e^2$$

SST: sum of squares total

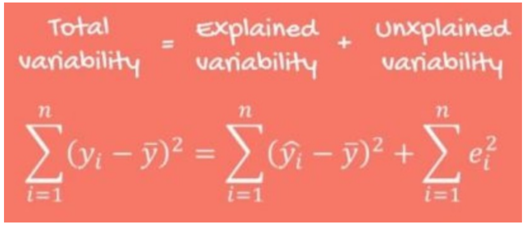
$$SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

SSR: sum of squares due to regression

$$SSR = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$$



$$SST = SSR + SSE$$
$$R^2 = 1 - SSE \,/\, SST$$



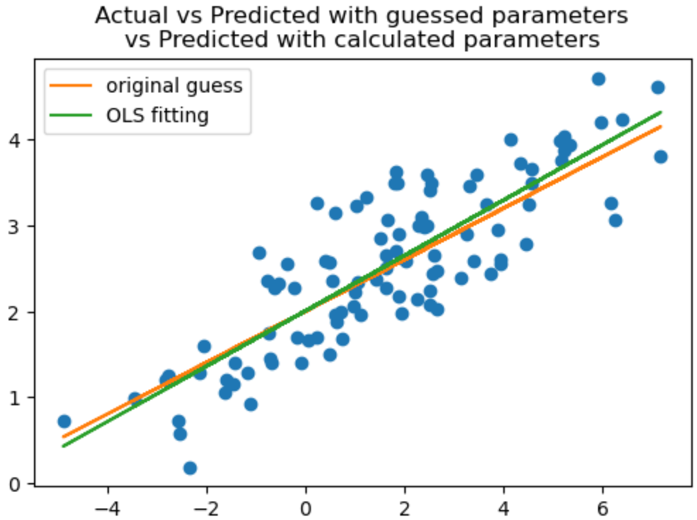
Total variability = Explained variability + Unexplained variability

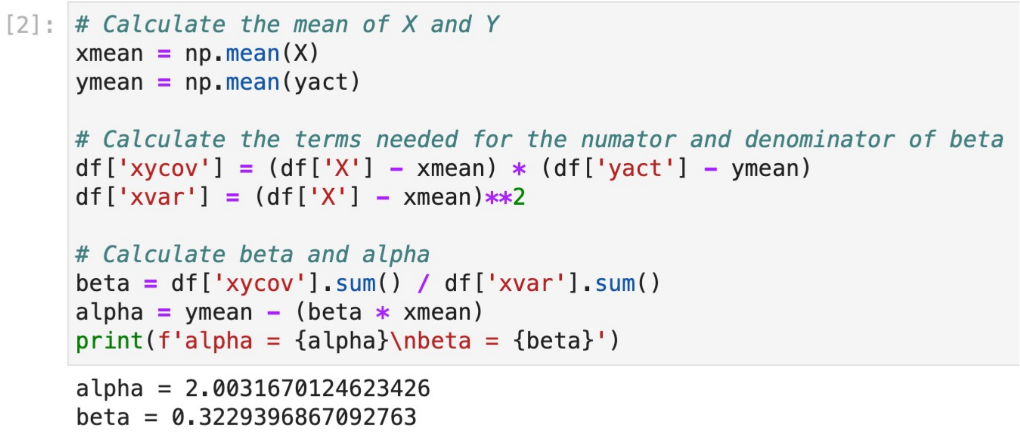
$$\sum_{i=1}^{n} (y_i - \bar{y})^2 = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^{n} e_i^2$$

https://en.wikipedia.org/wiki/Coefficient_of_determination
https://vitalflux.com/linear-regression-explained-python-sklearn-examples/

Simple linear regression | Example on model fitting

- Notebook1 for Data generation, colab: https://bit.ly/3CzOK4C
- Notebook2 for Least Squares fitting, colab: https://bit.ly/3nP1Vby
- Notebook3 for parameter evaluation, colab: https://bit.ly/3mvhPs7




```
[2]: # Calculate the mean of X and Y
     xmean = np.mean(X)
     ymean = np.mean(yact)

     # Calculate the terms needed for the numator and denominator of beta
     df['xycov'] = (df['X'] - xmean) * (df['yact'] - ymean)
     df['xvar'] = (df['X'] - xmean)**2

     # Calculate beta and alpha
     beta = df['xycov'].sum() / df['xvar'].sum()
     alpha = ymean - (beta * xmean)
     print(f'alpha = {alpha}\nbeta = {beta}')
```

```
alpha = 2.0031670124623426
beta = 0.3229396867092763
```

Simple linear regression | Example for diagnostic statistics

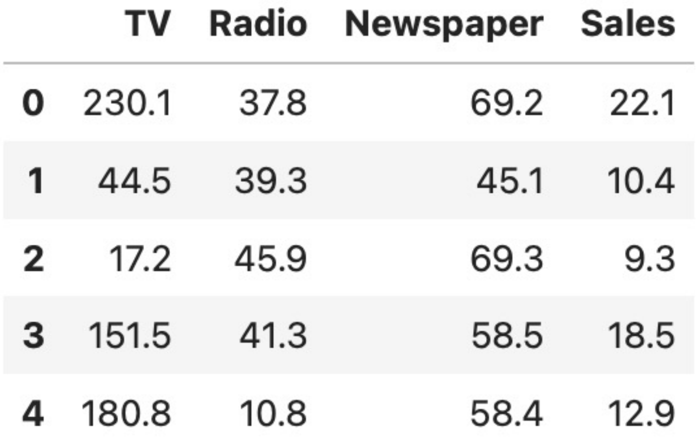
Data set: advertising.csv

- Output variable: Sales
- Input variables: advertising costs through TV, Radio, Newspaper

- n = 200 observations (rows)

- We fit a simple regression model:

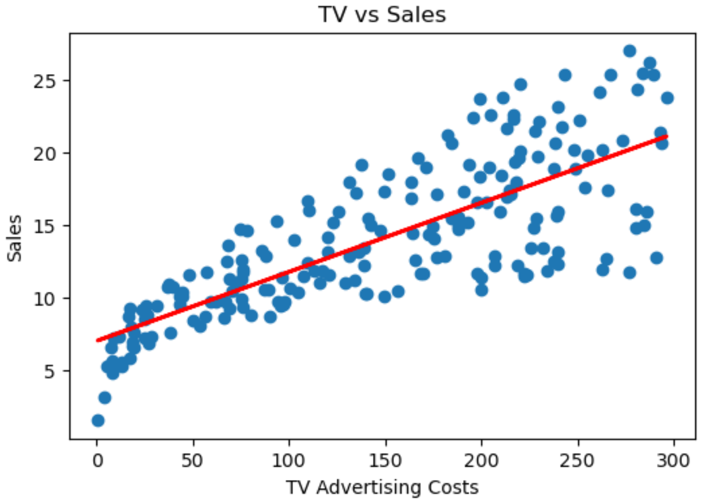
$$Sales = 7.032 + 0.047 * TV$$

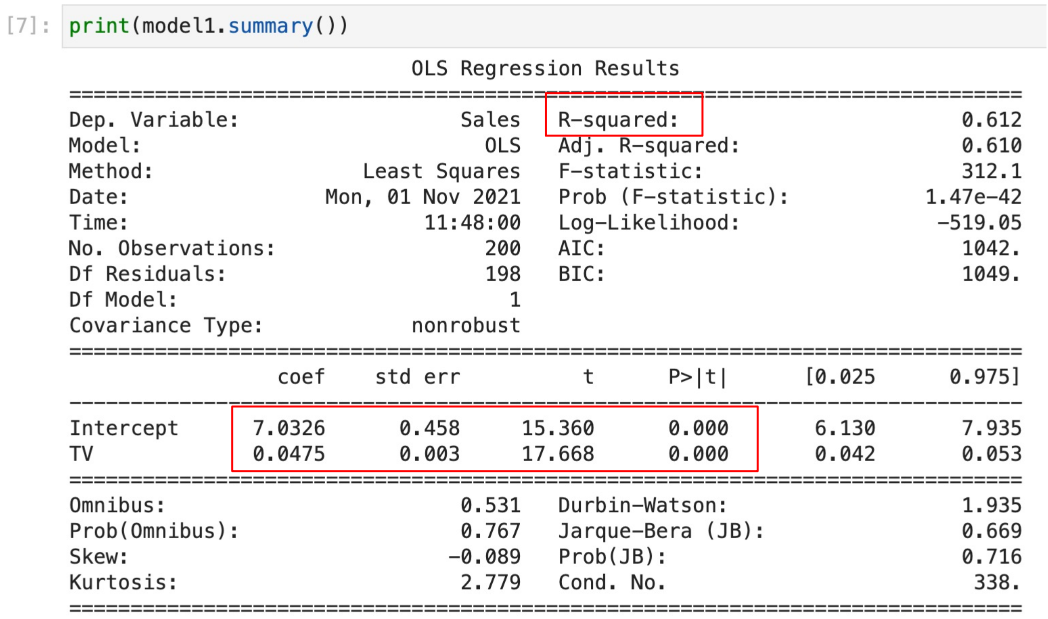


	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	9.3
3	151.5	41.3	58.5	18.5
4	180.8	10.8	58.4	12.9

Colab notebook: http://bit.ly/3byqbZG

Simple linear regression | Example for diagnostic statistics





```
[7]: print(model1.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Sales   R-squared:                       0.612
Model:                            OLS   Adj. R-squared:                  0.610
Method:                 Least Squares   F-statistic:                     312.1
Date:                Mon, 01 Nov 2021   Prob (F-statistic):           1.47e-42
Time:                        11:48:00   Log-Likelihood:                -519.05
No. Observations:                 200   AIC:                             1042.
Df Residuals:                     198   BIC:                             1049.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      7.0326      0.458     15.360      0.000       6.130       7.935
TV             0.0475      0.003     17.668      0.000       0.042       0.053
==============================================================================
Omnibus:                        0.531   Durbin-Watson:                   1.935
Prob(Omnibus):                  0.767   Jarque-Bera (JB):                0.669
Skew:                          -0.089   Prob(JB):                        0.716
Kurtosis:                       2.779   Cond. No.                         338.
==============================================================================
```

Simple linear regression | Example for diagnostic statistics

Coefficient of determination R^2=0.612

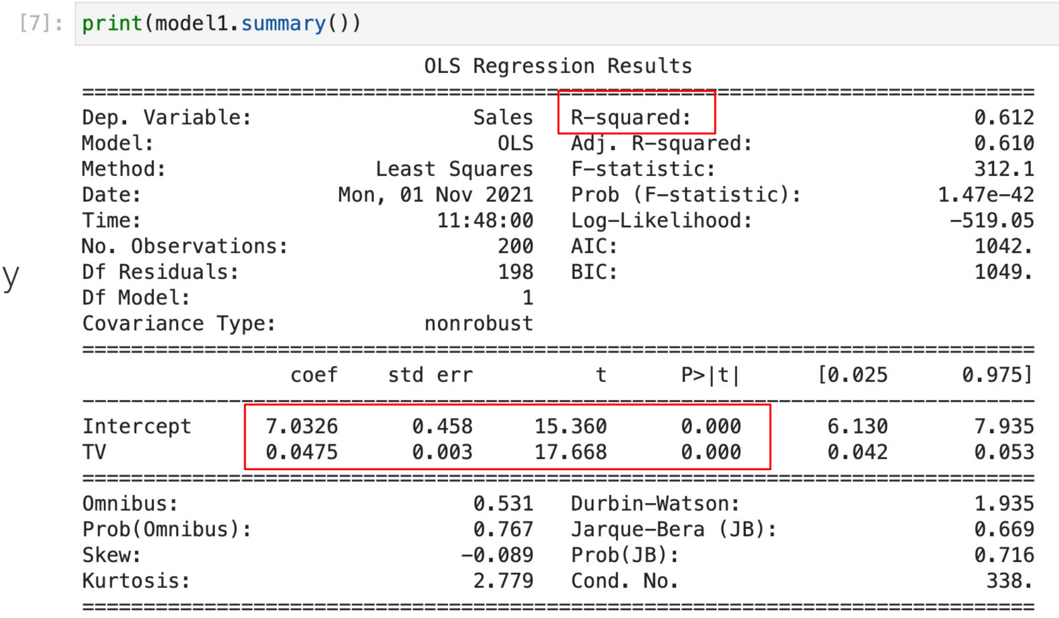
==> room for improvement

α =7.0326

- p-value for testing “α =0”: 0.000, the null hypothesis is rejected, and we conclude that the intercept coefficient is significantly non-zero.

slope β = 0.0475

- p-value for testing “β =0”: 0.000, the null hypothesis is rejected, and we conclude that the slope coefficient is significantly non-zero.
- As β is associated to the input variable TV, we conclude that the input variable TV has a significant impact on the output variable Sales.



```
[7]: print(model1.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Sales   R-squared:                       0.612
Model:                            OLS   Adj. R-squared:                  0.610
Method:                 Least Squares   F-statistic:                     312.1
Date:                Mon, 01 Nov 2021   Prob (F-statistic):           1.47e-42
Time:                        11:48:00   Log-Likelihood:                -519.05
No. Observations:                 200   AIC:                             1042.
Df Residuals:                     198   BIC:                             1049.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      7.0326      0.458     15.360      0.000       6.130       7.935
TV             0.0475      0.003     17.668      0.000       0.042       0.053
==============================================================================
Omnibus:                        0.531   Durbin-Watson:                   1.935
Prob(Omnibus):                  0.767   Jarque-Bera (JB):                0.669
Skew:                          -0.089   Prob(JB):                        0.716
Kurtosis:                       2.779   Cond. No.                         338.
==============================================================================
```

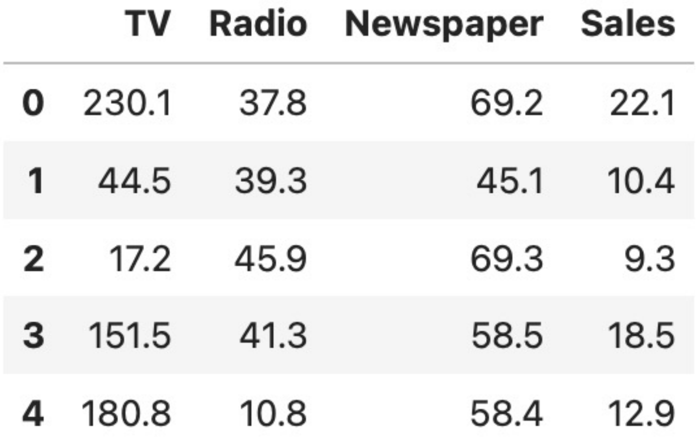
Part 2: multiple linear regression model

Multiple linear regression | example

Data set: advertising.csv

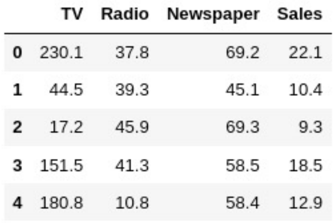
- Output variable: Sales
- Input variables: advertising costs through TV, Radio, Newspaper

- n = 200 observations (rows)



	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	9.3
3	151.5	41.3	58.5	18.5
4	180.8	10.8	58.4	12.9

Multiple linear regression | example



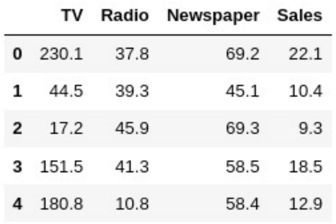
	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	9.3
3	151.5	41.3	58.5	18.5
4	180.8	10.8	58.4	12.9

- **Data set:** advertising.csv

- We target at regression model using more than one input variables. For example,
  - $Sales\ =\ a\ +\ b_1 * TV\ +\ b_2 * Radio$
  - $Sales\ =\ a\ +\ b_1 * TV\ +\ b_3 * Newspaper$
  - $Sales\ =\ a\ +\ b_1 * TV\ +\ b_2 * Radio\ +\ b_3 * Newspaper$

- We expect a better prediction of the output by using more predictors

Multiple linear regression | general methodology



	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	9.3
3	151.5	41.3	58.5	18.5
4	180.8	10.8	58.4	12.9

- Predict the output y using p input variables $x_1, x_2, \ldots, x_p$:

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

- The data set:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \text{ and } \begin{bmatrix} x_1^{(1)} & \cdots & x_1^{(p)} \\ x_2^{(1)} & \cdots & x_2^{(p)} \\ \vdots & \ddots & \vdots \\ x_1^{(1)} & \cdots & x_n^{(p)} \end{bmatrix}$$

- For each sample point i, the predicted output is:

$$\hat{y} = \alpha + \beta_1 x_i^{(1)} + \beta_2 x_i^{(2)} + \cdots + \beta_p x_i^{(p)}$$

- Least squares method: find the parameters such that the sum of squares

$$S(\alpha, \beta_1, \ldots, \beta_p) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \left(y_i - (\alpha + \beta_1 x_i^{(1)} + \cdots + \beta_p x_i^{(p)}) \right)^2$$

This can be done again by equating the partial derivatives to 0.

Multiple linear regression | the least squares estimators

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \text{ and } \begin{bmatrix} x_1^{(1)} & \cdots & x_1^{(p)} \\ x_2^{(1)} & \cdots & x_2^{(p)} \\ \vdots & \ddots & \vdots \\ x_1^{(1)} & \cdots & x_n^{(p)} \end{bmatrix}$$

- The sum of squares:

$$S\left(\alpha, \beta_1, \dots, \beta_p\right) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \left(y_i - (\alpha + \beta_1 x_i^{(1)} + \cdots + \beta_p x_i^{(p)})\right)^2$$

- Least sum of squares: solve

$$\frac{\partial S(\alpha, \beta_1, \dots, \beta_p)}{\partial \alpha} = 0, \quad \frac{\partial S(\alpha, \beta_1, \dots, \beta_p)}{\partial \beta_1} = 0, \quad \dots, \quad \frac{\partial S(\alpha, \beta_1, \dots, \beta_p)}{\partial \beta_p} = 0.$$

- The obtained values are the least squares estimators:

$$\hat{\alpha}, \hat{\beta}_1, \dots, \hat{\beta}_p$$

Multiple linear regression | the least squares estimators

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \text{ and } \begin{bmatrix} x_1^{(1)} & \cdots & x_1^{(p)} \\ x_2^{(1)} & \cdots & x_2^{(p)} \\ \vdots & \ddots & \vdots \\ x_1^{(1)} & \cdots & x_n^{(p)} \end{bmatrix}$$

- The sum of squares:

$$S(\alpha, \beta_1, \dots, \beta_p) = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \left(y_i - (\alpha + \beta_1 x_i^{(1)} + \dots + \beta_p x_i^{(p)}) \right)^2$$

- The difference of sum of squares is by definition: $SSE = S(\alpha, \beta_1, \dots, \beta_p)$

- Again, we have: $SST \ = \ SSR \ + \ SSE$

- The coefficient of determination is defined as before: $R^2 \ = \ SSR \ / \ SST$

Multiple Linear regression: Towards a wise selection of input variables

- Consider the advertising data set: with three available variables, TV, Radio, Newspaper in total we have 7 possible regression models:

- Which of these 7 regression models would be the best one?

- And, best one in which sense ?

1. Sales ~ TV
2. Sales ~ Newspaper
3. Sales ~ Radio
4. Sales ~ TV + Radio
5. Sales ~ TV + Newspaper
6. Sales ~ Newspaper + Radio
7. Sales ~ TV + Radio + Newspaper

We run into the complex question of selection of variables.

Important fact | The more variables, the bigger R^2

- Consider two regression models:
  - (I) $\hat{y} = \alpha + \beta_1 x_i$
  - (II) $\hat{y} = \alpha + \beta_1 x_i + \beta_2 x_2$
- The model (II) has one more variable x_2

- The SSE of Model (I) S_I is the minimum of over all possible values of (α, β_1)

$$S_I(\alpha, \beta_1) = \sum_{i=1}^{n} \left(y_i - (\alpha + \beta_1 x_i^{(1)})\right)^2$$

- The SSE of Model (II) S_{II} is the minimum of over all possible values of $(\alpha, \beta_1, \beta_2)$

$$S_{II}(\alpha, \beta_1, \beta_2) = \sum_{i=1}^{n} \left(y_i - (\alpha + \beta_1 x_i^{(1)} + \beta_2 x_i^{(2)})\right)^2$$

- Observations $S_I(\alpha, \beta_1) \ = S_{II}(\alpha, \beta_1, \beta_2 = 0)$, we get $S_{II} \ \leq S_I$,

$$R_{II}^2 = 1 - \frac{SSE_{II}}{SST} \geq 1 - \frac{SSE_I}{SST} = R_I^2$$

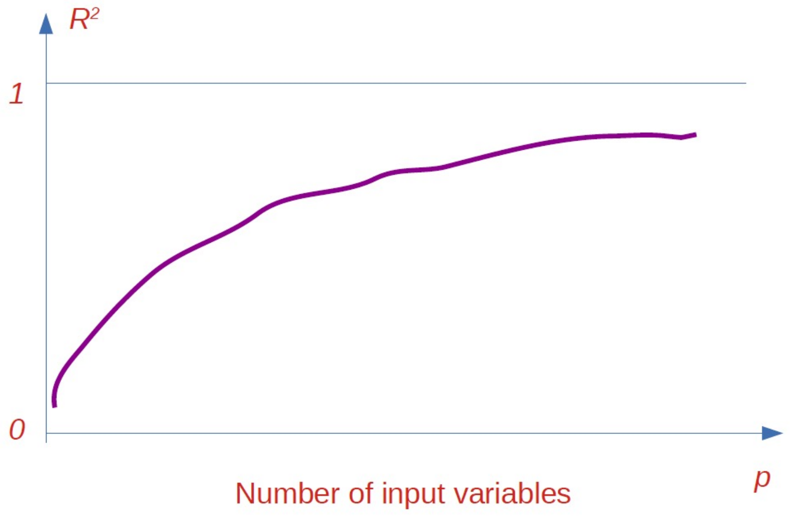
Multiple Linear regression | how many variables to use?

- Because of

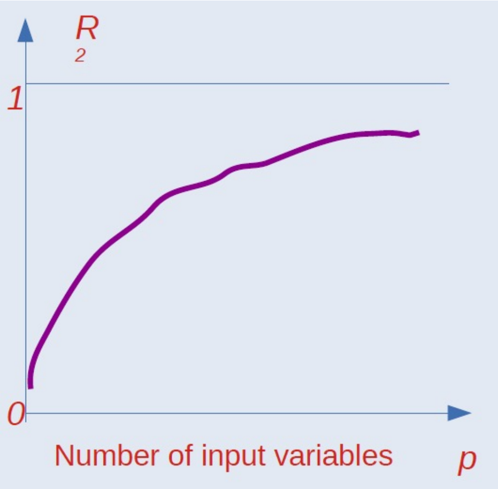
The more variables, the bigger R^2,

if we target solely at increasing R^2, we will finish up by incorporating all the variables available in the data set.

- Most likely this will result in a disastrous model (problem of overfitting – not generalisable to unobserved data)



Multiple Linear regression | how many variables to use?



	Model	R^2	Adjust-R^2	p-values for coefficients	F-statistic (p-value)	Error (= RSE / mean)
1	`Sales ~ TV`	0.612	0.610	0.000 (x 2)	312.1 (0.000)	23.2%
2	`Sales ~ TV + Newspaper`	0.646	0.642	0.000 (x 3)	179.6 (0.000)	22.2%
3	`Sales ~ TV + Radio`	0.897	0.896	0.000 (x 3)	859.7 (0.000)	12.0%
4	`Sales ~ TV + Newspaper + Radio`	0.897	0.896	0.000 (x4)	570.3 (0.000)	12.2%

RSE: residual standard error

$$RSE = \sqrt{SSE \ / \ (n - p - 1)}$$

Recommend | Use adjusted-R^2 to select input variables

- A useful criterion: instead of optimizing R^2, we chose a model, that is a subset of available variables such that the adjusted - R2 is maximum

$$R^2_{adj} = 1 - (1 - R^2)\frac{n-1}{n-p-1}$$

- So, when one increases p, the number of input variables, the reduction of $(1-R^2)$ is compensated by an increase in the factor (n-1) / (n-p+1).

- This will prevent from systematically picking a model with a large number of input variables p.

Recommend | Use adjusted-R^2 to select input variables

- Also keeping an eye on the following statistics:
  1) Individual regression coefficient should be significant: large *t*-statistic and small p value
  2) Global Fisher statistic should be large enough with small p-value

- Combined with the adjusted-R^2 selection, this will lead to a satisfactory selection of input variables to build a final regression model.

Advertising data: reach to a best fit

- From Model 1 to Model 2:
  - marginal increase of adjusted-R^2(+)
  - marginal decrease of Error (+)
  - big decrease of F-statistic (---)
- From Model 1 to Model 3:
  - significant increase of adjusted-R^2(++)
  - significant decrease of Error (++)
  - big increase of F-statistic (++)
- From Model 3 to Model 4:
  - no increase of adjusted-R^2(--)
  - marginal increase of Error (--)
  - big decrease of F-statistic (---)

	Model	R^2	Adjust-R^2	p-values for coefficients	F-statistic (p-value)	Error (= RSE / mean)
1	`Sales ~ TV`	0.612	0.610	0.000 (x 2)	312.1 (0.000)	23.2%
2	`Sales ~ TV + Newspaper`	0.646	0.642	0.000 (x 3)	179.6 (0.000)	22.2%
3	`Sales ~ TV + Radio`	0.897	0.896	0.000 (x 3)	859.7 (0.000)	12.0%
4	`Sales ~ TV + Newspaper + Radio`	0.897	0.896	0.000 (x4)	570.3 (0.000)	12.2%

Model (3) Sales ~ TV + Radio appears to be the best choice.

Colab notebook: http://bit.ly/2Y1GriP

Part 3: Basic intro for non-linear regression

Capturing non-linear relationship by data transformation

- Sometimes the output variable doesn't have a direct linear relationship with the predictor variable. They can have quadratic, exponential, logarithmic, or polynomial relationships.

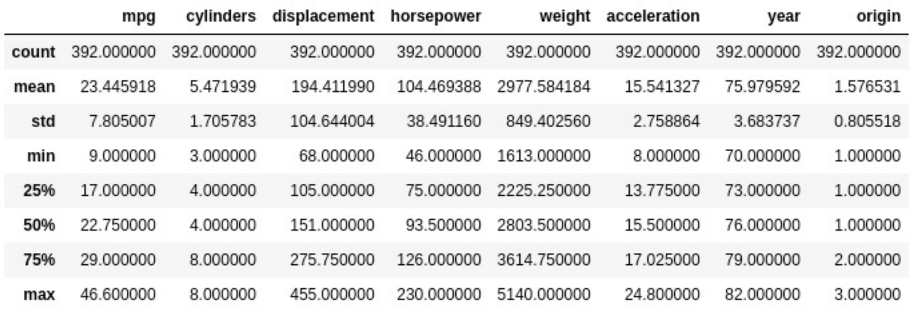
- In such cases, transforming the variable comes in very handy.

- The following is a rough guideline on how to spot and handle non-linear relationships:
  - Plot a scatter plot of the output variable with each of the predictor variables.
  - If the scatter plot assumes more or less a linear shape, then it is linearly related to the output variable.
  - If the scatter plot assumes a characteristic non-linear shape, then transform the variable by applying that function.

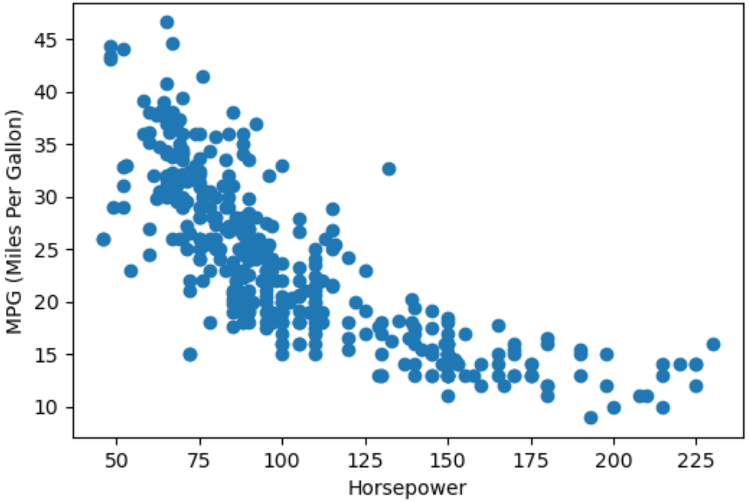
Capturing non-linear relationship | Example

Data: auto.csv

- Output = MPG (miles per gallon)
- Input variable = horsepower



	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin
count	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000
mean	23.445918	5.471939	194.411990	104.469388	2977.584184	15.541327	75.979592	1.576531
std	7.805007	1.705783	104.644004	38.491160	849.402560	2.758864	3.683737	0.805518
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.000000	4.000000	105.000000	75.000000	2225.250000	13.775000	73.000000	1.000000
50%	22.750000	4.000000	151.000000	93.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	275.750000	126.000000	3614.750000	17.025000	79.000000	2.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000	3.000000

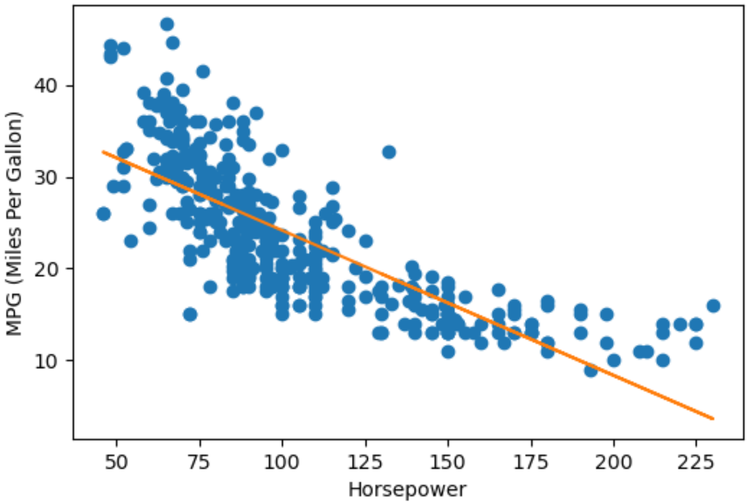


The relationship doesn't seem to have a linear shape

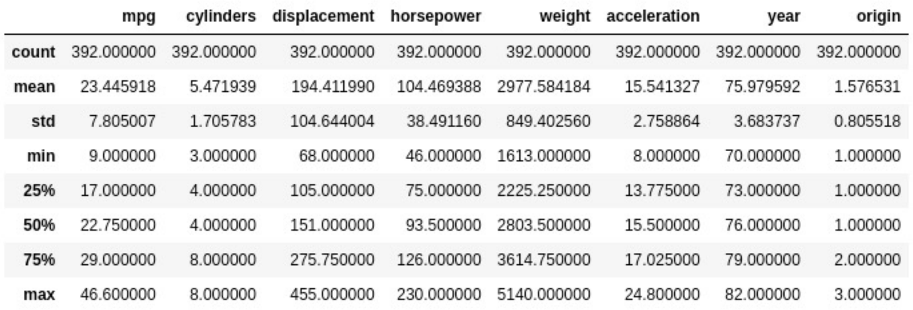
Capturing non-linear relationship | linear model

Data: auto.csv

- Output = MPG (miles per gallon)
- Input variable = horsepower



Not a very satisfactory fit.



	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin
count	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000
mean	23.445918	5.471939	194.411990	104.469388	2977.584184	15.541327	75.979592	1.576531
std	7.805007	1.705783	104.644004	38.491160	849.402560	2.758864	3.683737	0.805518
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.000000	4.000000	105.000000	75.000000	2225.250000	13.775000	73.000000	1.000000
50%	22.750000	4.000000	151.000000	93.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	275.750000	126.000000	3614.750000	17.025000	79.000000	2.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000	3.000000

Linear model fit:

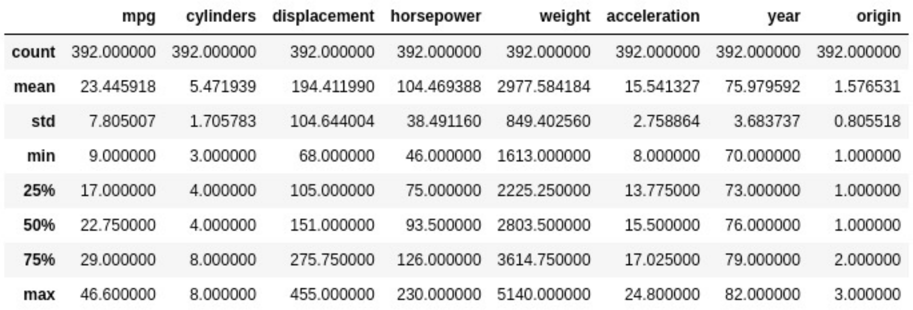
mpg = 39.93 - 0.1578 * horsepower

- ○ R^2 = 0.6059
- ○ RSE = 4.9058
- ○ Error = 20.92%

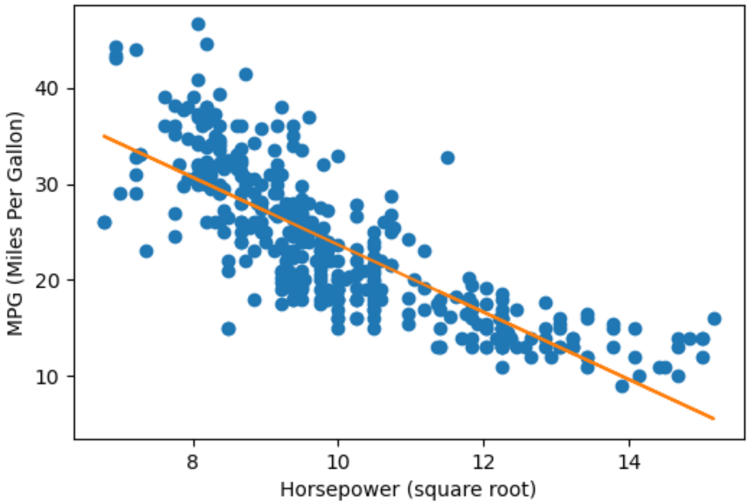
Capturing non-linear relationship | square-root transformation

Data: auto.csv

- Output = MPG (miles per gallon)
- Input variable = horsepower



	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin
count	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000
mean	23.445918	5.471939	194.411990	104.469388	2977.584184	15.541327	75.979592	1.576531
std	7.805007	1.705783	104.644004	38.491160	849.402560	2.758864	3.683737	0.805518
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.000000	4.000000	105.000000	75.000000	2225.250000	13.775000	73.000000	1.000000
50%	22.750000	4.000000	151.000000	93.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	275.750000	126.000000	3614.750000	17.025000	79.000000	2.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000	3.000000



Only a slight improvement.

Linear model fit with square-root transformed input:

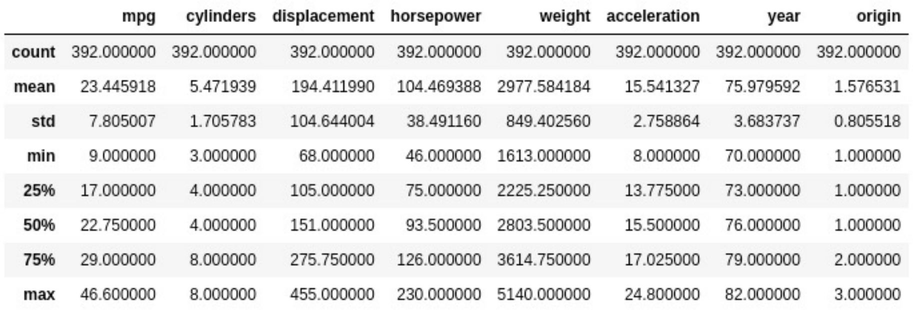
mpg = 39.93 - 0.1578 *(horsepower)1/2

- ○ R^2 = 0.6437 [previously, 0.6059]
- ○ RSE = 4.6648 [previously, 4.9058]
- ○ Error = 19.90% [previously, 20.92%]

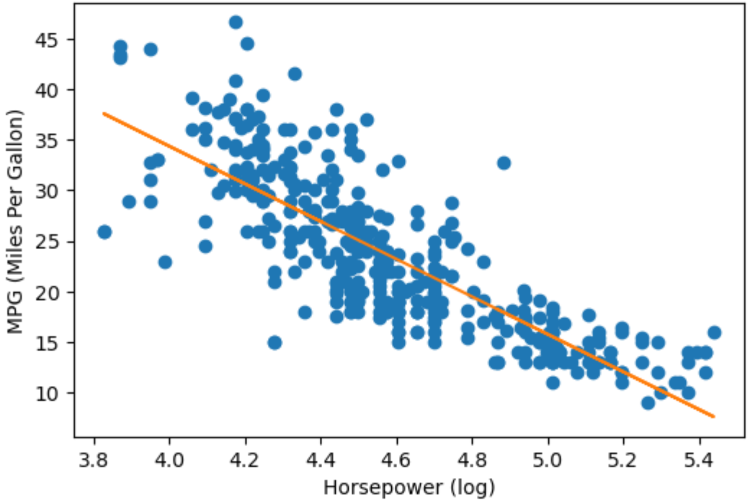
Capturing non-linear relationship | log transformation

Data: auto.csv

- Output = MPG (miles per gallon)
- Input variable = horsepower



	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin
count	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000
mean	23.445918	5.471939	194.411990	104.469388	2977.584184	15.541327	75.979592	1.576531
std	7.805007	1.705783	104.644004	38.491160	849.402560	2.758864	3.683737	0.805518
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.000000	4.000000	105.000000	75.000000	2225.250000	13.775000	73.000000	1.000000
50%	22.750000	4.000000	151.000000	93.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	275.750000	126.000000	3614.750000	17.025000	79.000000	2.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000	3.000000



Linear model fit with log-transformed input:

Mpg = 108.70 - 18.58 * log(horsepower)

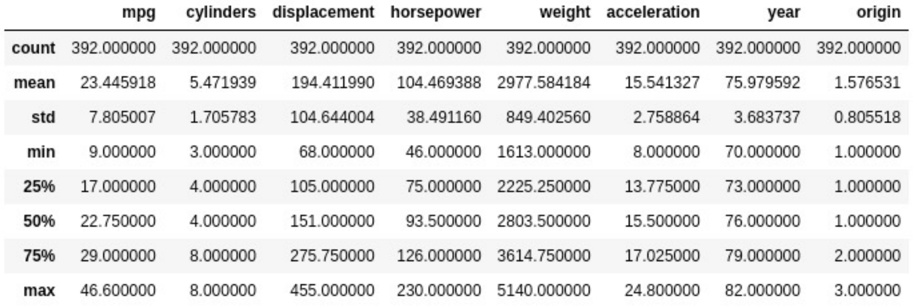
- ○ R^2 = 0.6683 [prev. 0.6059, 0.6437]
- ○ RSE = 4.5007 [prev. 4.9058, 4.6648]
- ○ Error = 19.19% [prev. 20.92%, 19.90%]

Again a small improvement only.

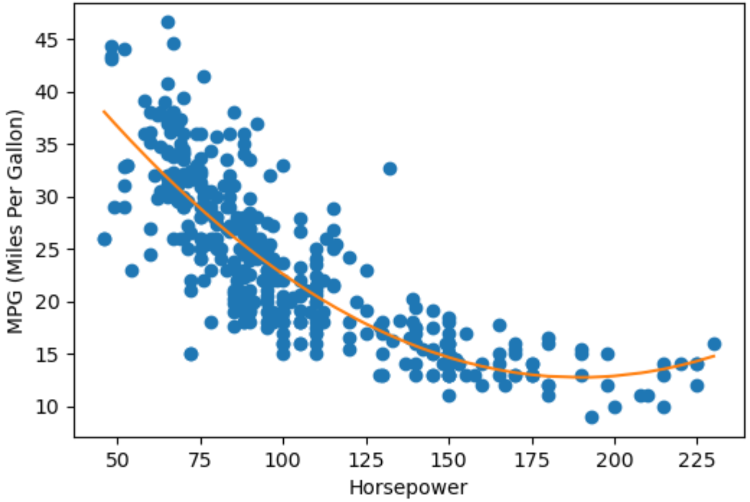
Capturing non-linear relationship | degree-2 polynomial transformation

Data: auto.csv

- Output = MPG (miles per gallon)
- Input variable = horsepower



	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin
count	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000
mean	23.445918	5.471939	194.411990	104.469388	2977.584184	15.541327	75.979592	1.576531
std	7.805007	1.705783	104.644004	38.491160	849.402560	2.758864	3.683737	0.805518
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.000000	4.000000	105.000000	75.000000	2225.250000	13.775000	73.000000	1.000000
50%	22.750000	4.000000	151.000000	93.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	275.750000	126.000000	3614.750000	17.025000	79.000000	2.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000	3.000000



We obtain a significant improvement.

Linear model fit with polynomial function of the input:

$$\text{Mpg} = 56.90 - 4.6662 * \text{horsepower} + 0.001 * \text{horsepower}^2$$

- ○ R2 = 0.6875 [prev. 0.6059, 0.6437, 0.6683]
- ○ RSE = 4.3739 [prev. 4.9058, 4.6648, 4.5007]
- ○ Error = 18.66% [prev. 20.92%, 19.90%, 19.19%]

Colab notebook: http://bit.ly/3jVdyMX

Summary

- Simple linear regression:
  - Linear relationship between input and output
  - Least squares estimators & and analytical solution exists

- Simple linear regression:
  - Least squares estimators & and analytical solution exists
  - How to select the combination of the input features (adjusted R^2)
  - Cross-validation (will introduce next lecture)

- Explore Non-linear relationship by transformation:
  - Visualization
  - Log or square root are common options
  - Higher degree of polynomial transformation – risk of overfitting

Resources & Acknowledgement

- IPython Notebook for this lecture note:
  - On Moodle
  - Also: https://github.com/huangyh09/foundation-data-science/

Other reference resources with acknowledgement:

- Chapter 4, Bruces & Gedeck, Practical Statistics for Data Science