Regression analysis

ANALYZING SURVEY DATA IN PYTHON

EbunOluwa Andrew
Data Scientist

datacamp

Regression analysis

- Understand the relationship between variables
- Utilized to predict a precise outcome
- Gauge influence of different independent variables on dependent variable
- Forecasts potential future opportunities and risks
- Reduces huge piles of raw data into actionable information
- Provides factual support for informed decisions

Linear regression using ordinary least squares (OLS) method

- Linear regression model
  - Assumes linear relationship between x and y variable
  - **y = m * x + b**
  - Ordinary Squares (OLS) Method
  - Sum((calculated-observed)^2) => minimized

[1] https://seeing-theory.brown.edu/regression-analysis/index.html

Loading data

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm
exercise_data = pd.read_csv('workout_survey_data.csv')
print(exercise_data.head())
```

```
| workout_minutes | calories_burned |
|-----------------|-----------------|
| 77              | 79.775152       |
| 21              | 23.177279       |
| 22              | 25.609262       |
| 20              | 17.857388       |
```

Define variables

x = independent variable y = dependent variable

```
x = exercise_data.minutes.tolist()
y = exercise_data.calories.tolist()
print(x,'\n',y)
```

```
| [77, 21, 22, 20, 36...                |
|---------------------------------------|
| [79.7, 23.1, 25.6, 17.8, 41.8... |
```

Survey data

workout_minutes	calories_burned
77	79.775152
21	23.177279
22	25.609262
20	17.857388
36	41.849864

Add constant term

```
x = sm.add_constant(x)
print (x)
```

- Tells model to fit a value for b

```
[[ 1. 77.]
 [ 1. 21.]
 [ 1. 22.]
 [ 1. 20.]
 [ 1. 36.]
 [ 1. 15.]
 [ 1. 62.]
 [ 1. 95.]
 [ 1. 20.]
 [ 1.  5.]
 [ 1.  4.]
 [ 1. 19.]
 [ 1. 96.]
```

Perform regression and fit

```
result = sm.OLS(y,x).fit()
print(result.summary())
```

OLS Regression Results

Dep. Variable:	y	R-squared:	0.990
Model:	OLS	Adj. R-squared:	0.990
Method:	Least Squares	F-statistic:	4945.
Date:	Sat, 17 Dec 2022	Prob (F-statistic):	4.47e-50
Time:	07:21:28	Log-Likelihood:	-123.40
No. Observations:	50	AIC:	250.8
Df Residuals:	48	BIC:	254.6
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.1552	0.830	0.187	0.852	-1.513	1.823
x1	1.0072	0.014	70.322	0.000	0.978	1.036

Omnibus:	0.856	Durbin-Watson:	1.793
Prob(Omnibus):	0.652	Jarque-Bera (JB):	0.860
Skew:	0.290	Prob(JB):	0.651
Kurtosis:	2.725	Cond. No.	117.

Retrieving m and b

OLS Regression Results

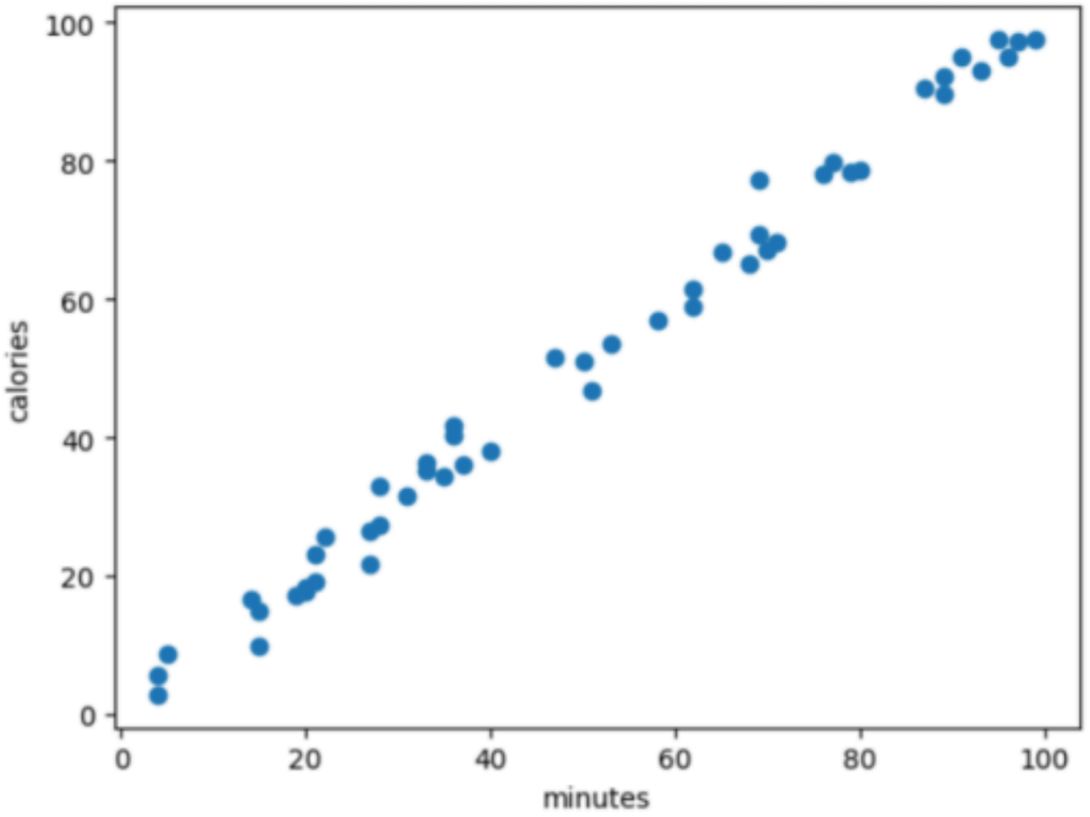
Dep. Variable:	y	R-squared:	0.990
Model:	OLS	Adj. R-squared:	0.990
Method:	Least Squares	F-statistic:	4945.
Date:	Sat, 17 Dec 2022	Prob (F-statistic):	4.47e-50
Time:	07:21:28	Log-Likelihood:	-123.40
No. Observations:	50	AIC:	250.8
Df Residuals:	48	BIC:	254.6
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.1552	0.830	0.187	0.852	-1.513	1.823
x1	1.0072	0.014	70.322	0.000	0.978	1.036

Omnibus:	0.856	Durbin-Watson:	1.793

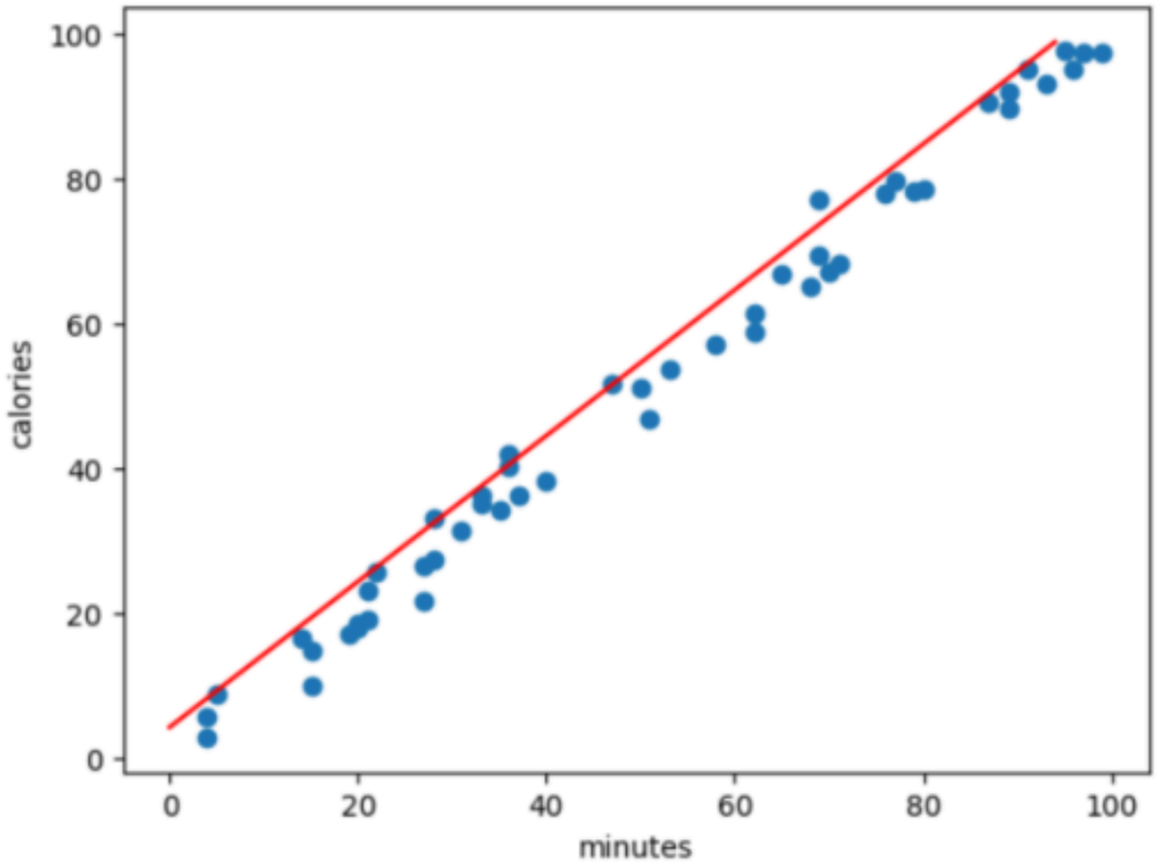
Plot original values

```
x = exercise_data.minutes.tolist()
y = exercise_data.calories.tolist()
plt.scatter(x,y)
plt.xlabel('minutes')
plt.ylabel('calories')
plt.show()
```



Plotting the regression line

```
max_x = exercise_data.minutes.max()
min_x = exercise_data.minutes.min()
x = np.arange(min_x, max_x, 1)
y = 1.0072*x + 0.1552
plt.plot(y, 'r')
plt.show()
```



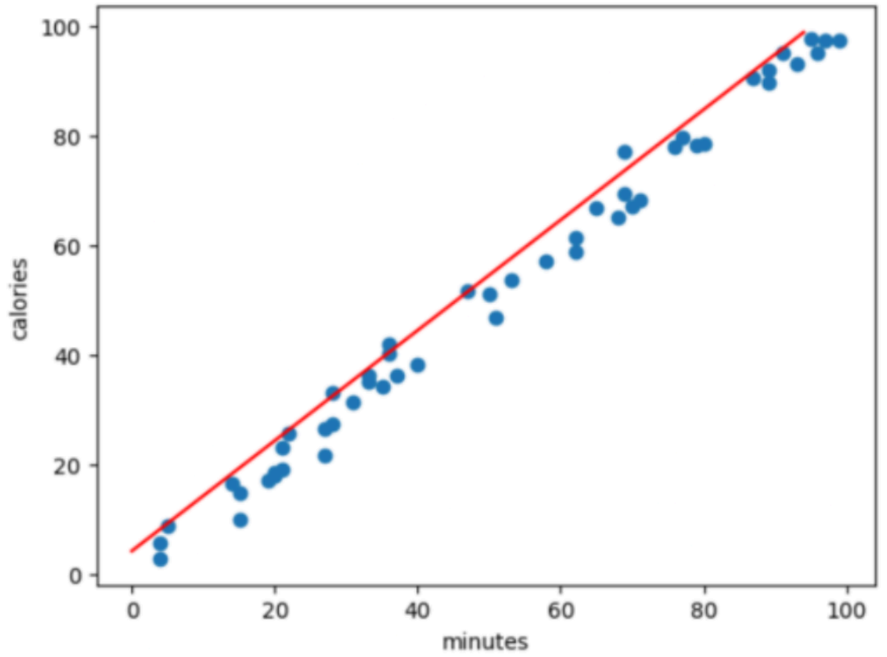
Predict response

```
y = 1.0072 * 30 + 0.1552
print(y)
```

```
30.3712
```

Linear regression pros and cons

- Pro
  - Performs well when data is linearly separable
- Con
  - Assumes linear relationship for non-linear cases



Let's practice!

ANALYZING SURVEY DATA IN PYTHON

Two sample t-test

ANALYZING SURVEY DATA IN PYTHON

EbunOluwa Andrew
Data Scientist

datacamp

Comparing agreeableness

```
group_a.agreeableness.mean()
```

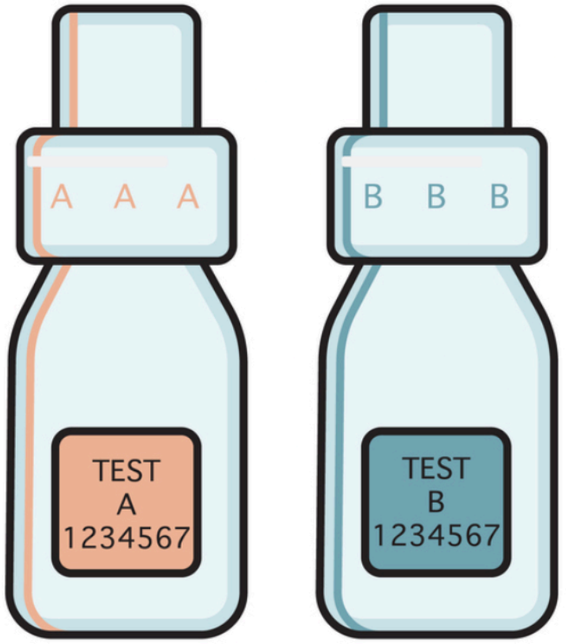
```
4.011701199563795
```

```
group_b.agreeableness.mean()
```

```
4.03669574700109
```

Define two sample t-test

- Examines whether the means of two independent groups are significantly different
- Determine whether differences are by chance



Assumptions for a two sample t-test

- Independent
- Normal distribution
  - Shapiro-Wilk test
  - `stats.shapiro()`
  - p-value > 0.05 -> normally distributed
- Equal variances
  - Levene test
  - `stats.levene()`
  - p-value > 0.05 -> equal variances

Survey results

group_a

userid	agreeableness
895	4.78
a06	3.40
e94	3.66
ee6	5.41
521	4.58
f4c	3.24
...

1 = Non-agreeable

group_b

userid	agreeableness
b7e	4.43
030	2.92
f91	4.01
36f	2.20
875	3.83
750	4.95
...

7 = Agreeable

Independent groups

Normally distributed groups

```
from scipy.stats import shapiro
import scipy.stats as stats

norm_A = stats.shapiro(
  group_a.agreeableness)
```

```
ShapiroResult(
statistic=0.997467577457428,
pvalue=0.16834689676761627)
```

```
from scipy.stats import shapiro
import scipy.stats as stats

norm_B = stats.shapiro(
  group_b.agreeableness)
```

```
ShapiroResult(
statistic=0.9987381100654602,
pvalue=0.7757995128631592)
```

Equal variances

```
import scipy.stats as stats


var_test = stats.levene(group_a.agreeableness, group_b.agreeableness)
```

```
LeveneResult(statistic=0.40492634057696597, pvalue=0.5246354858484796)
```

Assumptions checked

- Independent groups
  - no overlap of individuals
- Normally distributed groups
- Equal variances
  - no significant difference between the two variances

Two sample t-test with statsmodels

```
from scipy import stats
stats.ttest_ind(group_a.agreeableness, group_b.agreeableness)
```

Two sample t-test with statsmodels

```
Ttest_indResult(statistic=0.7746406648066304, pvalue=0.438651984836618 8)
```

Further analysis

```
group_a_mean = 4.011701199563795
```

```
group_b_mean = 4.03669574700109
```

Let's practice!

ANALYZING SURVEY DATA IN PYTHON

Chi-square test

ANALYZING SURVEY DATA IN PYTHON

EbunOluwa Andrew
Data Scientist

datacamp

Chi-square test

- Inferences about categorical variable distribution
  - Compares observed observations to expected observations

Chi-square test in survey analysis

- Decide relationship between two categorical variables of a population
- H_o = no relationship between variables
- H_a = relationship between variables
- P-value
  - if significant (<0.05), **reject** null hypothesis
  - if insignificant (>0.05), **accept** null hypothesis

Why use chi-square testing in survey analysis

- Input variables relevant to output variable
- Understand impact of different variables on population
- Check if differences are by chance or statistically significant

[1] Photo by Firmbee.com on Unsplash

Assumptions of chi-square test on survey analysis

- Both variables = categorical
- Sample randomly selected from population
- Sample size > 100
- Expected frequencies >=5

Survey data for chi-square analysis

pet_type	current_pets	time_spent	reduces_stress
dog	1	420	yes
dog	1	180	yes
dog	4	30	yes
dog	1	30	yes
dog	1	60	yes

Survey data for chi-square analysis

- Sample size >100
- Two categorical variables:
  - `pet_type`
  - `reduces_stress`

- H_o

  NO relationship between the type of pet owned by pet owners and their perceived reduced stress

- H_a

  relationship between the type of pet owned by pet owners and their perceived reduced stress

Steps of chi-square analysis on pet_survey in python

```
import pandas as pd
import scipy.stats as st
data = pd.read_csv('pet_survey.csv')
cross_table = pd.crosstab(data.reduces_stress, data.pet_type)
chi_analysis = st.chi2_contingency(cross_table)
print(chi_analysis)
```

```
|--------------------------|
| (67.7,                   |
| 1.9e-16,                 |
| 1,                       |
| array([[1767.0, 1825.0], |
| [2251.0, 2325.0]]))      |
```

Result and interpretation of pet_survey

- Frequencies >= 5
  - Valid results
- p-value < 0.05
  - **reject** null hypothesis
  - `pet_owned` and `reduces_stress` are related
- Type of pet owned has an effect on whether pet owners perceive stress reduction

chi_analysis		
67.7	<—	chi-squared value
1.90E-16	<—	p-value
1	<—	d.o.f
array([[1767.0, 1825.0], [2251.0, 2325.0]]))	<—	expected frequencies

Let's practice!

ANALYZING SURVEY DATA IN PYTHON

Congratulations

ANALYZING SURVEY DATA IN PYTHON

EbunOluwa Andrew
Data Scientist

Chapter one

- Different types of survey variables
- How to interpret inferential and descriptive statistics in surveys
- Visualization functions
  - `.scatter()`

Chapter two

- Create random sample from population data survey
- Account for sampling error
  - Stratified random sampling
  - Weighted sampling
  - Cluster sampling
- Visualization functions
  - `.pie()`
  - `.barh()`

Chapter three

- Difference between descriptive and inferential statistics
- Interpret:
  - meaning of different variables
  - central tendency, z-score
  - actionable steps

Chapter four

- Linear regression
- Two-sample t-test
- Chi-square test

The journey continues

Thank you

ANALYZING SURVEY DATA IN PYTHON