**Simple Linear Regression: Graded Quiz**

**LATEST SUBMISSION GRADE**

100%

1.Question 1

Which of the following evaluation metrics can be used to evaluate a model while modeling a continuous output variable?

**1 / 1 point**



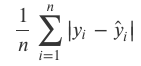
Accuracy



**Mean Absolute Error**

**Correct**

Correct! Linear Regression outputs continuous values. So we need an evaluation metric that compares continuous values. **Mean Absolute Error** (MAE) is the mean of the absolute value of the errors:





Logloss



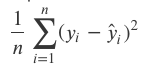
AUC-ROC



**Mean Squared Error**

**Correct**

Correct! Linear Regression outputs continuous values. So we need an evaluation metric that compares continuous values. **Mean Squared Error** (MAE) is the mean of the absolute value of the errors:





Precision

2.Question 2

***true*** and ***pred*** are two lists of true and predicted response values from a simple linear regression model. Using scikit-learn, how would you calculate the root mean squared error to evaluate the response values?

**1 / 1 point**





1

metrics.mean\_absolute\_error(true, pred)





1

**np.sqrt(metrics.mean\_squared\_eror(true, pred))**



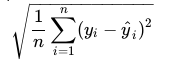


1

metrics.mean\_squared\_eror(true, pred)

**Correct**

Correct! The mathematical form of the RMSE evaluation metric is:



3.Question 3

For a pandas DataFrame **df**, what function is used to return the first five rows of the DataFrame?

**1 / 1 point**





1

df.info()





1

**df.head()**





1

df.tail()

**Correct**

Correct! This function, by default, returns the first five rows of the DataFrame. It is useful for quickly testing if your object has the right type of data in it. It can also return the first *n*rows when specified as an argument. Eg: The code below returns the first 10 rows from **df**.



1

df.head(10)

4.Question 4

A DataFrame, **df**, contains three columns: TV, Radio, Newspaper. Which of the following options drops the Radio column inplace?

**2 / 2 points**





1

df.drop(['TV','Newspaper'], axis=1, inplace=True)





1

**df.drop(['Radio'], axis=1, inplace=True)**

**Correct**

Good job! The first argument specifies the columns to be dropped. The *axis=1* argument drops the columns from **df**. Lastly, when *inplace=True*, the operation is performed inplace.





1

df.drop(['Radio'], axis=0, inplace=True)





1

df.drop(['Radio'], axis=0, inplace=False)

5.Question 5

Which of the following Seaborn functions did we use to plot a univariate distribution of observations?

**1 / 1 point**





1

sns.jointplot()





1

sns.distplot()

**Correct**

Correct! In Task 4, we used this function to plot the distribution of each column in the **advert** DataFrame.



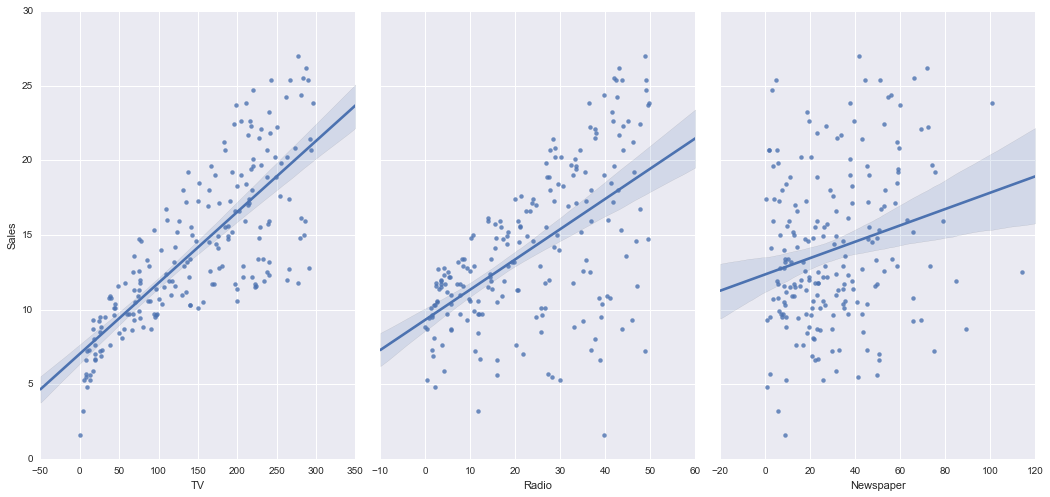


1

sns.kdeplot()

6.Question 6

The image below visualizes the relationship between the features *TV, Radio, Newspaper* and the response, *Sales*, using scatter plots. Each scatter plot is fit with a linear regression model.



Select all the conclusions that can be drawn from the above plots.

**1 / 1 point**



**There is a strong relationship between TV ads and Sales.**

**Correct**

Correct! The plot on the left shows a strong, positive, linear association between TV ads and Sales with a few outliers.



**There is a weak relationship between Radio ads and Sales.**

**Correct**

Correct! There is a weak, positive, linear association between Radio ads and Sales with many outliers.



**There is a very weak to no relationship between Newspaper ads and Sales**

**Correct**

Correct! There is a weak, non linear relationship between Newspaper ads and Sales.

7.Question 7

There are multiple features (explanatory variables) in a simple linear regression model.

**1 / 1 point**



True



False

**Correct**

Correct! A simple linear regression, is a linear regression model with a single feature (explanatory variable).

8.Question 8

For the simple linear regression model below, what terms correspond to the feature coefficient and y-intercept respectively?

https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/z5Wu5dFGEemkuwrX1d3ObA_f88503d1591c177035f199934beef1f2_Screenshot-from-2019-09-07-13-38-15.png?expiry=1592092800000&hmac=MDbNFz9OWJhLaya4pddZ9Sh_tFQMP4Y_D813XhNpD_4

**1 / 1 point**



https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/WpKgq9FIEemVGg7IEE0Tpg_87fe3cbae0ee5fa786177b1bb72eb0cb_Screenshot-from-2019-09-07-13-50-19.png?expiry=1592092800000&hmac=i6ZJDAXHtqIQrqX2239r2McQpW-8w8RcIKc6V_B7GEc



https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/iOtoBNFIEemgqgoxEn8lOg_e83b7fbf7c5fe587d6bee1613f3a3765_Screenshot-from-2019-09-07-13-51-30.png?expiry=1592092800000&hmac=-k9g2z0IlXRCwVA69C6-A85s-H-XN5sko9Hte_A0afk



https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/D2SbO9FJEem1wRICcqBpcA_6f2c9bc4095774d37fa91ba97c18fcce_Screenshot-from-2019-09-07-13-55-06.png?expiry=1592092800000&hmac=eCagIHp4QIkHoUnDs4y4c2CyDxqJcJIM-xFAFjQrUYc

**Correct**

Correct!

9.Question 9

The first line of code below shows some data being split into training and test sets for features *X* and response *y*. What two objects returned from the *train\_test\_split* function need to be passed as arguments to *linreg.fit(*( )?



1

2

3

4

5

6

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y)

from sklearn.linear\_model import LinearRegression

linreg = LinearRegression()

linreg.fit()

**1 / 1 point**



*X\_train* and *X\_test*



***X\_train* and *y\_train***



*X\_test* and *y\_test*

**Correct**

Correct Scikit-learn's linear regression estimator is fit on the training features and response.