| **Scoring** | **Function** | **Comment** |
| --- | --- | --- |
| **Classification** |  |  |
| ‘accuracy’ | [**metrics.accuracy\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html#sklearn.metrics.accuracy_score) |  |
| ‘average\_precision’ | [**metrics.average\_precision\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html#sklearn.metrics.average_precision_score) |  |
| ‘f1’ | [**metrics.f1\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html#sklearn.metrics.f1_score) | for binary targets |
| ‘f1\_micro’ | [**metrics.f1\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html#sklearn.metrics.f1_score) | micro-averaged |
| ‘f1\_macro’ | [**metrics.f1\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html#sklearn.metrics.f1_score) | macro-averaged |
| ‘f1\_weighted’ | [**metrics.f1\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html#sklearn.metrics.f1_score) | weighted average |
| ‘f1\_samples’ | [**metrics.f1\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html#sklearn.metrics.f1_score) | by multilabel sample |
| ‘neg\_log\_loss’ | [**metrics.log\_loss**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.log_loss.html#sklearn.metrics.log_loss) | requires predict\_proba support |
| ‘precision’ etc. | [**metrics.precision\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_score.html#sklearn.metrics.precision_score) | suffixes apply as with ‘f1’ |
| ‘recall’ etc. | [**metrics.recall\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.recall_score.html#sklearn.metrics.recall_score) | suffixes apply as with ‘f1’ |
| ‘roc\_auc’ | [**metrics.roc\_auc\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html#sklearn.metrics.roc_auc_score) |  |
| **Clustering** |  |  |
| ‘adjusted\_rand\_score’ | [**metrics.adjusted\_rand\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.adjusted_rand_score.html#sklearn.metrics.adjusted_rand_score) |  |
| **Regression** |  |  |
| ‘neg\_mean\_absolute\_error’ | [**metrics.mean\_absolute\_error**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_error.html#sklearn.metrics.mean_absolute_error) |  |
| ‘neg\_mean\_squared\_error’ | [**metrics.mean\_squared\_error**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_error.html#sklearn.metrics.mean_squared_error) |  |
| ‘neg\_median\_absolute\_error’ | [**metrics.median\_absolute\_error**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.median_absolute_error.html#sklearn.metrics.median_absolute_error) |  |
| ‘r2’ | [**metrics.r2\_score**](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html#sklearn.metrics.r2_score) |  |

## Simple Linear Regression

* Simple linear regression is an approach for predicting a **quantitative response** using a **single feature** (or "predictor" or "input variable")
* It takes the following form:
* y=β0+β1xy=β0+β1x

What does each term represent?

* yy is the response
* xx is the feature
* β0β0 is the intercept
* β1β1 is the coefficient for x
* β0β0 and β1β1 are called the **model coefficients**

## Using the Model for Prediction

Let's say that there was a new market where the TV advertising spend was **$50,000**. What would we predict for the Sales in that market?

y=β0+β1xy=β0+β1x

y=7.032594+0.047537×50

**Confidence in our Model**

**Question:** Is linear regression a high variance/low bias model, or a low variance/high bias model?

**Answer:**

* Low variance/high bias
* Under repeated sampling, the line will stay roughly in the same place (low variance)
  + But the average of those models won't do a great job capturing the true relationship (high bias)
* Note that low variance is a useful characteristic when you don't have a lot of training data

A closely related concept is **confidence intervals**

* Statsmodels calculates 95% confidence intervals for our model coefficients, which are interpreted as follows:
  + If the population from which this sample was drawn was **sampled 100 times**
    - Approximately **95 of those confidence intervals** would contain the "true" coefficient

**Hypothesis Testing and p-values**

**Steps for Hypothesis Testing**

1. Start with a **null hypothesis** and an **alternative hypothesis** (that is opposite the null)
2. Then, you check whether the data supports **rejecting the null hypothesis** or **failing to reject the null hypothesis**
   * "failing to reject" the null is not the same as "accepting" the null hypothesis
   * The alternative hypothesis may indeed be true, except that you just don't have enough data to show that

**Conventional hypothesis test**

* **null hypothesis:**
  + There is no relationship between TV ads and Sales
    - $\beta\_1$ equals zero
* **alternative hypothesis:**
  + There is a relationship between TV ads and Sales
    - $\beta\_1$ is not equal to zero

**Testing hypothesis**

* Reject the null
  + There is a relationship
  + If the 95% confidence interval **does not include zero**
* Fail to reject the null
  + There is no relationship
  + If the 95% confidence interval **includes zero**

**p-value**

* Represents the probability that the coefficient is actually zero

**Interpreting p-values**

* If the 95% confidence interval **does not include zero**
  + p-value will be **less than 0.05**
  + Reject the null
  + There is a relationship
* If the 95% confidence interval **includes zero**
  + p-value for that coefficient will be **greater than 0.05**
  + Fail to reject the null
  + There is no relationship

**How Well Does the Model Fit the data?**

To evaluate the overall fit of a linear model, we use the **R-squared** value

* R-squared is the **proportion of variance explained**
  + It is the proportion of variance in the observed data that is explained by the model, or the reduction in error over the **null model**
    - The null model just predicts the mean of the observed response, and thus it has an intercept and no slope
* R-squared is between 0 and 1
  + Higher values are better because it means that more variance is explained by the model.

**Diagram explanation**

* **Blue line** explains some of the variance in the data (R-squared=0.54)
* **Green line** explains more of the variance (R-squared=0.64)
* **Red line** fits the training data even further (R-squared=0.66)

## Multiple Linear Regression

Simple linear regression can easily be extended to include multiple features. This is called **multiple linear regression**:

$y = \beta\_0 + \beta\_1x\_1 + ... + \beta\_nx\_n$

Each $x$ represents a different feature, and each feature has its own coefficient. In this case:

$y = \beta\_0 + \beta\_1 \times TV + \beta\_2 \times Radio + \beta\_3 \times Newspaper$

Let's estimate these coefficients:

**Model Evaluation Metrics for Regression**

For classification problems, we have only used classification accuracy as our evaluation metric. What metrics can we used for regression problems?

**Mean Absolute Error** (MAE) is the mean of the absolute value of the errors:

1n∑i=1n|yi−y^i|1n∑i=1n|yi−y^i|

**Mean Squared Error** (MSE) is the mean of the squared errors:

1n∑i=1n(yi−y^i)21n∑i=1n(yi−y^i)2

**Root Mean Squared Error** (RMSE) is the square root of the mean of the squared errors:

1n∑i=1n(yi−y^i)2−−−−−−−−−−−−√