

Perform a Linear Regression - Wolfram Mathematica

Выполнение линейной регрессии

Одной из самых распространенных статистических моделей является модель линейной регрессии. Линейная модель предсказывает значение зависимой переменной от линейной комбинации предикторных (прогнозируемых) переменных или функций предикторных переменных. В Mathematica, функция [LinearModelFit](#) возвращает объект, содержащий данные подбора по точкам для модели линейной регрессии, и позволяет легко извлекать результаты и проводить диагностику.

Сформируем набор данных:

In[1]:=

```
data =  
  Table[{3 + i + RandomReal[{-3, 7}], i + RandomReal[{-2, 5}]}, {i, 1, 1000}]
```

Воспользуемся функцией [LinearModelFit](#) для создания линейной модели для заданного набора данных:

In[2]:=

```
model = LinearModelFit[data, x, x]
```

Out[2]=

```
FittedModel[-1.75033 + 0.849824 x]
```

Извлечем функциональную форму модели:

In[4]:=

```
model["BestFit"]
```

Out[4]=

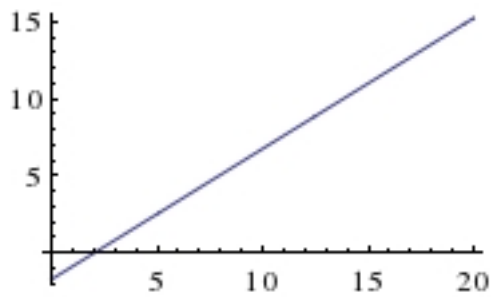
```
-1.75033 + 0.849824 x
```

Построим график функциональной формы модели:

In[5]:=

```
Plot[model["BestFit"], {x, 0, 20}]
```

Out[5]=

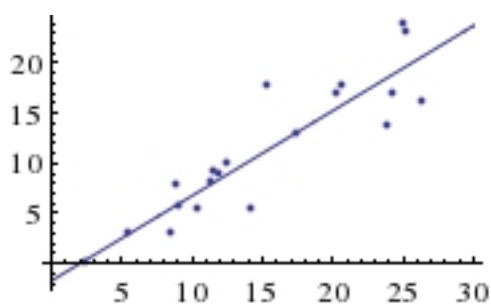


Отобразим данные и линию наилучшего соответствия:

In[6]:=

```
Show[ListPlot[data], Plot[model["BestFit"], {x, 0, 30}]]
```

Out[6]=



Получим информацию по оценке параметров выполненной регрессии:

In[7]:=

```
model ["ParameterTable"]
```

Out[7]=

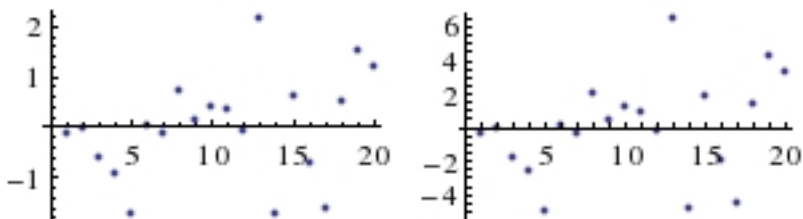
	Estimate	Standard Error	t-Statistic	P-Value
1	-1.75033	1.63732	-1.06902	
x	0.849824	0.0974775	8.71815	

Извлечем и отобразим на графике нормализованные и наблюдаемые невязки:

In[8]:=

```
{sr, fr} = model [{"StandardizedResiduals", "FitResiduals"}];
{{ListPlot[sr], ListPlot[fr]} // GraphicsGrid
```

Out[9]=

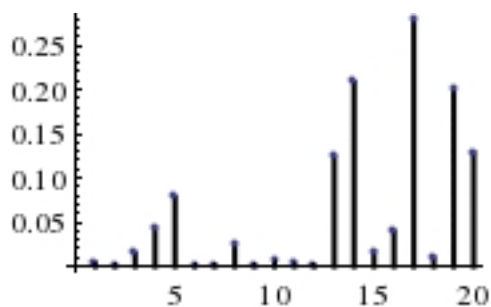


Построим график расстояний Кука:

In[10]:=

```
ListPlot[model["CookDistances"],  
  Filling → Axis, FillingStyle → Thick, PlotStyle → Thick]
```

Out[10]=

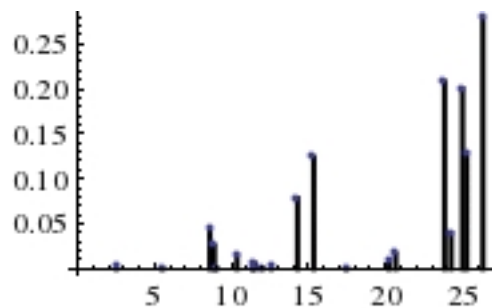


Как вариант, построим график расстояний Кука в зависимости от предикторного значения:

In[11]:=

```
ListPlot[Transpose[{data[[All, 1]], model["CookDistances"]}],  
  Filling → Axis, FillingStyle → Thick, PlotStyle → Thick]
```

Out[11]=



В предыдущих примерах были показаны лишь несколько свойств, поддерживаемых функцией [LinearModelFit](#) ; список доступных свойств намного обширнее:

In[12]:=

```
model["Properties"]
```

Out[12]=

{AdjustedRSquared, AIC, ANOVATable, ANOVATableDegreesOfFreedom, ANOVATableFStatistics, ANOVATableMeanSquares, ANOVATablePValues, ANOVATableSumsOfSquares, BasisFunctions, BetaDifferences, BestFit, BestFitParameters, BIC, CatcherMatrix, CoefficientOfVariation, CookDistance, CorrelationMatrix, CovarianceMatrix, CovarianceRatios, Data, DesignMatrix, DurbinWatsonD, EigenstructureTable, EigenstructureTableEigenvalues, EigenstructureTableEntries, EigenstructureTableIndexes, EigenstructureTablePartitions, EstimatedVariance, FitDifferences, FitResiduals, Function, FVarianceRatios, HatDiagonal, MeanPredictionBand, MeanPredictionConfidenceIntervals, MeanPredictionConfidenceIntervalTableEntries, MeanPredictionErrors, ParameterConfidenceIntervals, ParameterConfidenceIntervalTable, ParameterConfidenceIntervalTableEntries, ParameterConfidenceRegion, ParameterErrors, ParameterPValues, ParameterTable, ParameterTableEntries, ParameterTStatistics, PartialSumOfSquares, PredictedResponse, Properties, Response, RSquared, SequentialSumOfSquares, SingleDeletionVariances, SinglePredictionBand, SinglePredictionConfidenceIntervals, SinglePredictionConfidenceIntervalTableEntries, SinglePredictionErrors, StandardizedResiduals, StudentizedResiduals, VarianceInflationFactors