In[1]:=
```
ClearAll["Global`*"]
```

Метод наименьших квадратов

Математическая модель

1. Постановка задачи

Для заданного набора экспериментальных данных { X[i], Y[i] }, i = 1,..., n, определить функциональную зависимость Y = f[X], наилучшим образом описывающая экспериментальные данные. Фукциональная зависимость f называется моделью для экспериментальных данных или функцией регрессии.
Обычно в качестве модели выбирают достаточно простые функциональные зависимости, например,
линейная по переменной модель f[X] = a[0] X + a[1],
квадратичная модель f[X] = a[0] X^2 + a[1] X + a[2],
полиномиальная модель f[X] = a[0] X^m + a[1] X^{m-1} + ... + a[m-1] X +a[m], m < n,
степенная модель f[X] = a[1]*$X^{a[0]}$
показательная f[X] = a[1] *$a[0]^X$,
гиперболическая f[x] = a[1] + $\frac{a[0]}{X}$,
обощенно гиперболическая f[X] = a[0] + $\frac{a[1]}{X^{a[2]}}$.
Все эти модели включают некоторые неизвестные параметры a[0], ..., a[m]. Количество этих параметров m должно быть существенно меньше, чем общее количество n экспериментальных данных.
Для выбранной модели можно ввести ошибку приближения или ошибку аппроксимации в каждой точке исходных данных { err[1], err[2], ..., err[n] }:
err[i] = Y[i] - f[X[i]]], i = 1,...,n.
Для определения качества аппроксимации необходимо ввести некоторый критерий качества приближения - функция потерь. Один из наиболее популярных критериев - сумма квадратов ошибок :
S = $\sum_{i=1}^{n}(\mathrm{err}[i])^2 = (\mathrm{err}[1])^2 + (\mathrm{err}[2])^2 + \ldots + (\mathrm{err}[n])^2$.
Эта величина S = S(a[0],...,a[m]) зависит от параметров модели a[0],a[1],...,a[m].
Основная задача: определить значения параметров a[0],a[1],...,a[m], при которых сумма квадратов ошибок принимает наименьшее значение. Определение параметров по этому критерию называется Метод Наименьших Квадратов (МНК).

2. Метод наименьших квадратов

2.1. Метод наименьших квадратов для линейной модели

Для линейной модели f[x] = a[0] X + a[1] сумма квадратов ошибок S равна
S = $\sum_{i=1}^{n}(\mathrm{err}[i])^2 = \sum_{i=1}^{n}(Y[i] - f[X])^2 = \sum_{i=1}^{n}(Y[i] - a[0] * X[i] - a[1])^2$,

то есть зависит от двух переменных S=S[a[0], a[1]].
Минимальное значение этой величины достигается в точке, где градиент функции S равен нулю:
grad(S[a[0], a[1]]) = 0,
что дает линейную систему двух уравнений относительно a[0] и a[1]:

$$\begin{cases} 2\left(a[0]\sum_{i=1}^{n} X[i]^2 + a[1]\times\sum_{i=1}^{n} X[i] - \sum_{i=1}^{n} X[i]\times Y[i]\right) & = 0 \\ 2\left(a[0]\times\sum_{i=1}^{n} X[i] + n\,a[1] - \sum_{i=1}^{n} Y[i]\right) & = 0 \end{cases}$$

Обозначим

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X[i],\ \overline{Y} = \frac{1}{n}\sum_{i=1}^{n} Y[i],\ \overline{X^2} = \frac{1}{n}\sum_{i=1}^{n} X[i]^2,\ \overline{X*Y} = \frac{1}{n}\sum_{i=1}^{n} X[i]*Y[i].$$

Тогда система для определения наилучших значений a[0],a[1] примет следующий вид

$$\begin{cases} a[0]\,n\,\overline{X^2} + a[1]\,n\,\overline{X} & = n\,\overline{X\,Y} \\ a[0]\,n\,\overline{X} + n\,a[1] & = n\,\overline{Y} \end{cases}$$

или, поделив на n, получим

$$\begin{cases} a[0]\,\overline{X^2} + a[1]\,\overline{X} & = \overline{X*Y} \\ a[0]\,\overline{X} + a[1] & = \overline{Y} \end{cases}$$

Эта система двух линейных уравнений с двумя неизвестными (нормальная система для МНК) легко решается и ее решение {aopt[0], aopt[1]} называется оптимальным решением по методу наименьших квадратов.

2.2. Метод наименьших квадратов для остальных простейших моделей с двумя параметрами

Степенная модель f[X] = a[1]*$X^{a[0]}$.
Для этой модели критерий суммы квадратов равен
$S = \sum_{i=1}^{n} (\mathrm{err}[i])^2 = \sum_{i=1}^{n} (Y[i] - f[X])^2 = \sum_{i=1}^{n} \left(Y[i] - a[1]*X[i]^{a[0]}\right)^2$,
то есть зависит от двух переменных S=S[a[0], a[1]].
Использование стандартных метод дифференцирования приводят к большим сложностям. Поэтому для нахождения минимуму обычно используют численные методы, основанные на методе наискорейшего спуска.

Показательная модель f[X] = a[1] *$a[0]^X$.
Для этой модели критерий суммы квадратов равен
$S = \sum_{i=1}^{n} (\mathrm{err}[i])^2 = \sum_{i=1}^{n} (Y[i] - f[X])^2 = \sum_{i=1}^{n} \left(Y[i] - a[1]*a[0]^{X[i]}\right)^2$,
то есть зависит от двух переменных S=S[a[0], a[1]].
Использование стандартных метод дифференцирования приводят к большим сложностям. Поэтому для нахождения минимуму обычно используют численные методы, основанные на методе наискорейшего спуска.

Гиперболическая модель a[1] + $\frac{a[0]}{X}$.
Для этой модели параметры входят в виде линейной функции, поэтому можно использовать стандартную процедуру, вычисление градиента и решение системы нормальных уравнений.

3. Оценка качества построенной оптимальной модели

Существует много различных критериев оценки качества построенной модели. Рассмотрим только два полезных на практике.

Введем обозначение $\hat{Y}$ [i] - значение, предсказанное в соотвествии с моделью $\hat{Y}$ [i] = Y[i] - f[X[i]] = Y[i] - (aopt[0] X[i] + aopt[1]). Критерии оценки качества обычно выражаются через них.

3.1. Коэффициент детерминации R^2.

Обозначим SSE = $\sum_{i=1}^{n}\left(Y[i]-\hat{Y}[i]\right)^2$ - сумма квадратов отклонений (Sum of Squared Errors) и SST = $\sum_{i=1}^{n}\left(Y[i]-\overline{Y}\right)^2$ - общая сумма квадратов, отклонения Y[i] от среднего значения (Sum of Squared Total).
Тогда коэффициент детерминации R^2 равен:
$R^2 = 1-\frac{SSE}{SST}$. Чем ближе коэффициент детеминацияя к 1, тем лучше качество приближения.

3.2. Средняя относительная ошибка в процентах (MAPE = Mean absolute percentage error):

Этот критерий вычисляется по формуле
MAPE = $\frac{1}{n}\sum_{i=1}^{n}\frac{\left|Y[i]-\hat{Y}[i]\right|}{|Y[i]|}$,
чем меньше значение MAPE тем лучше качество приближения.

4. Генерация исходных данных.

Для построения исходных данных необходимо запустить код, приведенный в вариантах заданий.

5. Использование стандартных функций WM.

Построение функциональных зависимостей для заданного набора данных может осуществляться с помощью стандартной функции
FindFit [данные, модель, параметры, переменная].
Дополнительно для решения задачи можно использовать функции
LinearModelFit[...] и NonlinearModelFit[...].

Варианты заданий

Исходный код для генерации исходных данных data для обработки. В этом коде необходимо заменить номер_варианта на свой вариант задания. Числа npoints и p выбрать в таблице соотвествии с номером варианта

```
SeedRandom[номер_варианта];
data = Table[{i, p * (i + RandomReal[{0, npoints}])}, {i, 1, npoints}]
```

In[2]:=
```
Needs["GeneratorVariants1`"]
gridVar[{{"npoints", 10, 14, 1}, {"p", 1, 35, 1}}, 35]
```

Out[3]=

Вариант	npoints	p
1	10	1
2	11	2
3	12	3
4	13	4
5	14	5
6	10	6
7	11	7
8	12	8
9	13	9
10	14	10
11	10	11
12	11	12
13	12	13
14	13	14
15	14	15
16	10	16
17	11	17
18	12	18
19	13	19
20	14	20
21	10	21
22	11	22
23	12	23
24	13	24
25	14	25
26	10	26
27	11	27
28	12	28
29	13	29
30	14	30
31	10	31
32	11	32
33	12	33
34	13	34
35	14	35

Пример решения задачи

1. Исходные данные (вариант 35)

In[4]:=
```
npoints = 14
p = 35
```

Out[4]= 14

Out[5]= 35

Построение исходного набора данных

In[6]:=
```
SeedRandom[35];
data = Table[{i, p * (i + RandomReal[{0, npoints}])}, {i, 1, npoints}]
```

Out[7]=
```
{{1, 195.278}, {2, 413.238}, {3, 241.814}, {4, 297.835}, {5, 399.894},
 {6, 537.373}, {7, 371.448}, {8, 356.627}, {9, 577.297}, {10, 671.979},
 {11, 423.869}, {12, 603.441}, {13, 863.965}, {14, 705.427}}
```

Независимы и зависимые переменые

In[8]:=
```
X = data[[All, 1]]
Y = data[[All, 2]]
```

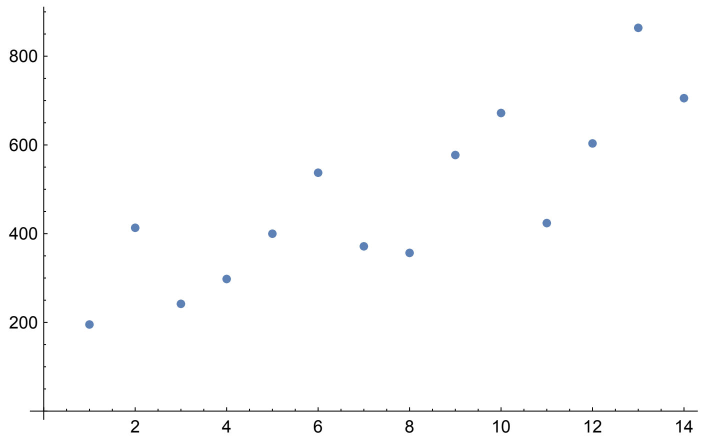
Out[8]=
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14}
```

Out[9]=
```
{195.278, 413.238, 241.814, 297.835, 399.894, 537.373, 371.448,
 356.627, 577.297, 671.979, 423.869, 603.441, 863.965, 705.427}
```

Графическое изображение исходных данных

In[10]:=
```
ListPlot[data]
```

Out[10]=



2. Построение моделей

2.1. Линейная модель f1 = a[0] X + a[1]

In[11]:=
```
f1[x_] = a[0] x + a[1]
```

Out[11]=
```
x a[0] + a[1]
```

Сумма квадратов ошибок - функция потерь

In[12]:=
```
S1 = Total[(Y - f1[X])^2]
```

Out[12]=

$$(705.427 - 14\,a[0] - a[1])^2 + (863.965 - 13\,a[0] - a[1])^2 + (603.441 - 12\,a[0] - a[1])^2 + (423.869 - 11\,a[0] - a[1])^2 + (671.979 - 10\,a[0] - a[1])^2 + (577.297 - 9\,a[0] - a[1])^2 + (356.627 - 8\,a[0] - a[1])^2 + (371.448 - 7\,a[0] - a[1])^2 + (537.373 - 6\,a[0] - a[1])^2 + (399.894 - 5\,a[0] - a[1])^2 + (297.835 - 4\,a[0] - a[1])^2 + (241.814 - 3\,a[0] - a[1])^2 + (413.238 - 2\,a[0] - a[1])^2 + (195.278 - a[0] - a[1])^2$$

Градиент функции потерь по переменным { a[0],a[1] }

In[13]:=
```
grad = D[S1, {{a[0], a[1]}}] // Simplify
```
Out[13]=
```
{2030. (-57.6771 + a[0] + 0.103448 a[1]), 210. (-63.4237 + a[0] + 0.133333 a[1])}
```

Система нормальных уравнений для определения оптимального значения параметров

In[14]:=
```
Thread[grad == 0] // MatrixForm
```
Out[14]//MatrixForm=

$$\begin{pmatrix} 2030.\ (-57.6771 + a[0] + 0.103448\, a[1]) == 0 \\ 210.\ (-63.4237 + a[0] + 0.133333\, a[1]) == 0 \end{pmatrix}$$

Решение этой системы

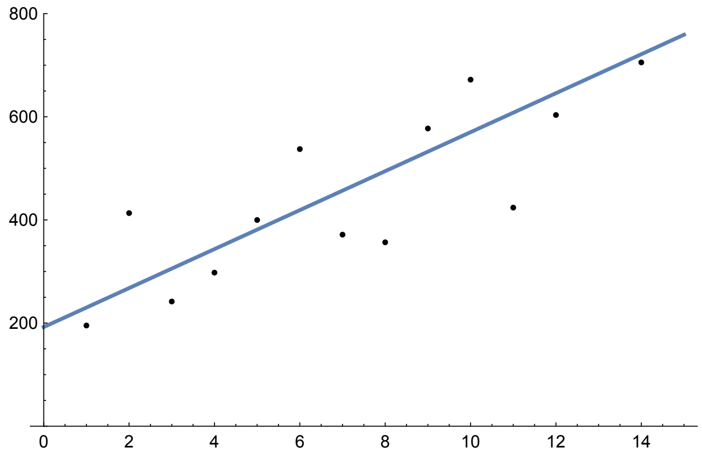
In[15]:=
```
{{aopt[0], aopt[1]}} = SolveValues[Thread[grad == 0], {a[0], a[1]}]
```
Out[15]=
```
{{37.785, 192.29}}
```

Оптимальная линейная модель

In[16]:=
```
f1opt[x_] = aopt[0] x + aopt[1]
```
Out[16]=
```
192.29 + 37.785 x
```

Совместный график исходных данных и построенной оптимальной функции

In[17]:=
```
Plot[f1opt[x], {x, 0, 15}, PlotRange → {0, 800}, Epilog → Point[data]]
```
Out[17]=



Вычислим критерии качества - коэффициент детерминации R^2 и MAPE.

In[18]:=
```
SSE1 = Total[(Y - f1opt[X])^2]
```
Out[18]=
```
150 046.
```

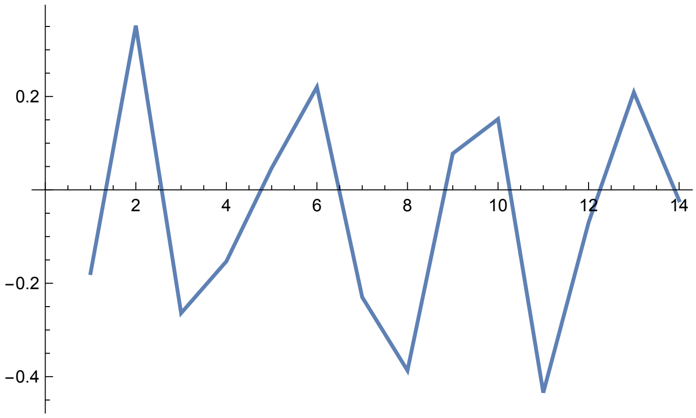
In[19]:=
```
SST1 = Total[(Y - Mean[Y])^2]
```
Out[19]=
```
474 849.
```

```
In[20]:= r2[1] = 1 - SSE1 / SST1
         r2[1] = Round[r2[1], 0.001]
Out[20]=
         0.684014
Out[21]=
         0.684

In[22]:= MAPE[1] = Mean[Abs[Y - f1opt[X]] / Abs[Y]]
         MAPE[1] = Round[MAPE[1], 0.001]
Out[22]=
         0.199687
Out[23]=
         0.2
```

Ошибки аппроксимации в каждой из точек

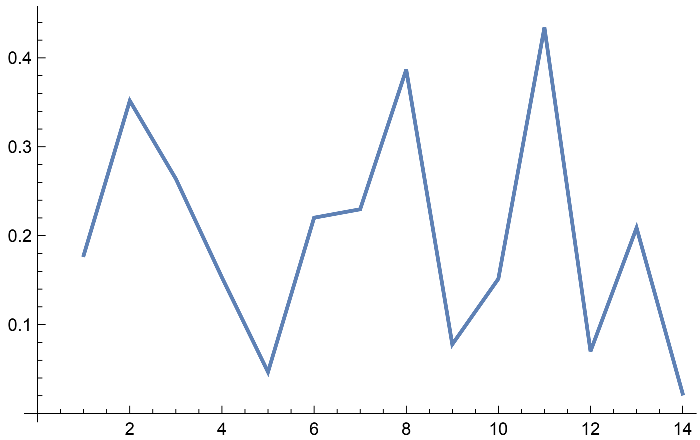
```
In[24]:= ListPlot[(Y - f1opt[X]) / Y, Joined → True]
Out[24]=
```



Относительная ошибка в каждой из точек

```
In[25]:= Abs[(Y - f1opt[X]) / Y]
Out[25]=
         {0.178191, 0.351802, 0.263967, 0.153089, 0.046711, 0.220282, 0.229741,
          0.3868, 0.0778491, 0.15155, 0.43423, 0.0700466, 0.208886, 0.0224731}

In[26]:= ListPlot[Abs[(Y - f1opt[X]) / Y], Joined → True]
Out[26]=
```



2.2. Степенная модель f = a[1]*$X^{a[0]}$

In[27]:=
```
f2[x_] = b[1] * x ^ b[0]
```
Out[27]=

$x^{b[0]}\ b[1]$

Сумма квадратов ошибок - функция потерь

In[28]:=
```
S2 = Total[(Y - f2[X]) ^ 2]
```
Out[28]=

$$(195.278 - b[1])^2 + (413.238 - 2^{b[0]}\ b[1])^2 + (241.814 - 3^{b[0]}\ b[1])^2 + (297.835 - 4^{b[0]}\ b[1])^2 + (399.894 - 5^{b[0]}\ b[1])^2 + (537.373 - 6^{b[0]}\ b[1])^2 + (371.448 - 7^{b[0]}\ b[1])^2 + (356.627 - 8^{b[0]}\ b[1])^2 + (577.297 - 9^{b[0]}\ b[1])^2 + (671.979 - 10^{b[0]}\ b[1])^2 + (423.869 - 11^{b[0]}\ b[1])^2 + (603.441 - 12^{b[0]}\ b[1])^2 + (863.965 - 13^{b[0]}\ b[1])^2 + (705.427 - 14^{b[0]}\ b[1])^2$$

В виду сложности функции потерь можно использовать стандартные функции определения минимума, а не вычислять градиент и искать критические точки.

In[29]:=
```
min2 = FindMinimum[S2, {b[0], b[1]}]
```
Out[29]=

{166 084., {b[0] → 0.533644, b[1] → 169.067}}

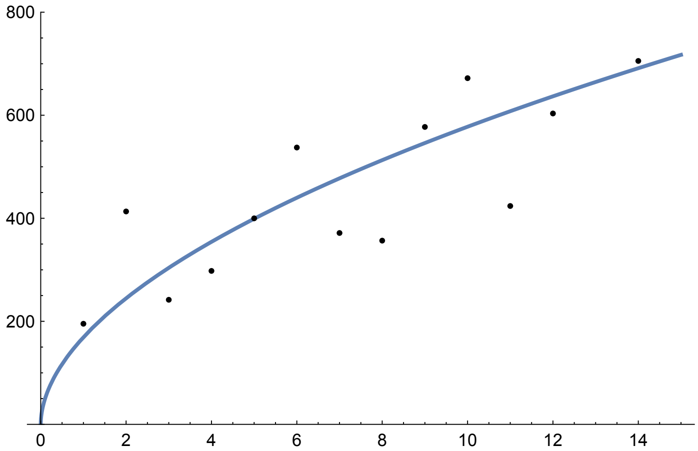
Оптимальная степенная модель

In[30]:=
```
f2opt[x_] = f2[x] /. min2[[2]]
```
Out[30]=

$169.067\ x^{0.533644}$

Совместный график исходных данных и построенной оптимальной функции

In[31]:=
```
Plot[f2opt[x], {x, 0, 15}, PlotRange → {0, 800}, Epilog → Point[data]]
```
Out[31]=



Вычислим критерии качества - коэффициент детерминации R^2 и MAPE.

In[32]:=
```
SSE2 = Total[(Y - f2opt[X]) ^ 2]
```
Out[32]=

166 084.

In[33]:=
```
SST2 = Total[(Y - Mean[Y])^2]
```
Out[33]=
```
474849.
```

In[34]:=
```
r2[2] = 1 - SSE2 / SST2
r2[2] = Round[r2[2], 0.001]
```
Out[34]=
```
0.650238
```
Out[35]=
```
0.65
```

In[36]:=
```
MAPE[2] = Mean[Abs[Y - f2opt[X]] / Abs[Y]]
MAPE[2] = Round[MAPE[2], 0.001]
```
Out[36]=
```
0.20212
```
Out[37]=
```
0.202
```

2.3. Показательная модель f = a[1]*a[0]^x

In[38]:=
```
f3[x_] = c[1] * c[0] ^ x
```
Out[38]=

$c[0]^x\, c[1]$

Сумма квадратов ошибок - функция потерь

In[39]:=
```
S3 = Total[(Y - f3[X])^2]
```
Out[39]=

$$(195.278 - c[0] \times c[1])^2 + \left(413.238 - c[0]^2 c[1]\right)^2 + \left(241.814 - c[0]^3 c[1]\right)^2 + \left(297.835 - c[0]^4 c[1]\right)^2 + \left(399.894 - c[0]^5 c[1]\right)^2 + \left(537.373 - c[0]^6 c[1]\right)^2 + \left(371.448 - c[0]^7 c[1]\right)^2 + \left(356.627 - c[0]^8 c[1]\right)^2 + \left(577.297 - c[0]^9 c[1]\right)^2 + \left(671.979 - c[0]^{10} c[1]\right)^2 + \left(423.869 - c[0]^{11} c[1]\right)^2 + \left(603.441 - c[0]^{12} c[1]\right)^2 + \left(863.965 - c[0]^{13} c[1]\right)^2 + \left(705.427 - c[0]^{14} c[1]\right)^2$$

В виду сложности функции потерь можно использовать стандартные функции определения минимума.

In[40]:=
```
min3 = FindMinimum[S3, {c[0], c[1]}]
```
Out[40]=
```
{144589., {c[0] → 1.08422, c[1] → 246.121}}
```

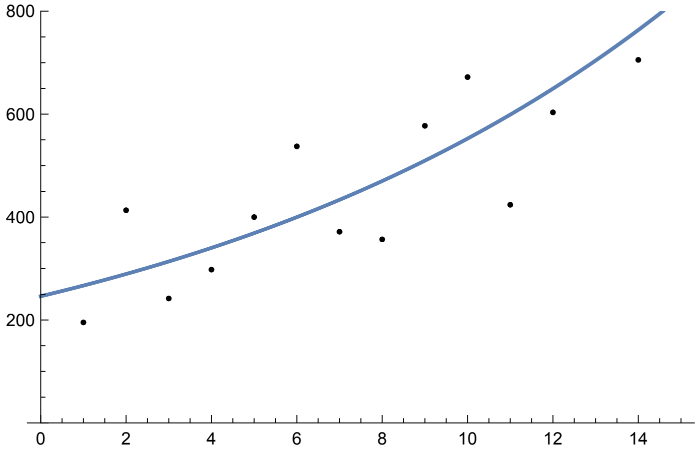
Оптимальная показательная модель

In[41]:=
```
f3opt[x_] = f3[x] /. min3[[2]]
```
Out[41]=

246.121×1.08422^x

Совместный график исходных данных и построенной оптимальной функции

In[42]:=
```
Plot[f3opt[x], {x, 0, 15}, PlotRange → {0, 800}, Epilog → Point[data]]
```
Out[42]=



Вычислим критерии качества - коэффициент детерминации R^2 и MAPE.

In[43]:=
```
SSE3 = Total[(Y - f3opt[X]) ^2]
```
Out[43]=
```
144 589.
```

In[44]:=
```
SST3 = Total[(Y - Mean[Y]) ^2]
```
Out[44]=
```
474 849.
```

In[45]:=
```
r2[3] = 1 - SSE3 / SST3
r2[3] = Round[r2[3], 0.001]
```
Out[45]=
```
0.695506
```
Out[46]=
```
0.696
```

In[47]:=
```
MAPE[3] = Mean[Abs[Y - f3opt[X]] / Abs[Y]]
MAPE[3] = Round[MAPE[3], 0.001]
```
Out[47]=
```
0.212588
```
Out[48]=
```
0.213
```

2.4. Гиперболическая модель $f = a[1] + \frac{a[0]}{x}$

In[49]:=
```
f4[x_] = d[1] + d[0] / x
```
Out[49]=

$$\frac{d[0]}{x} + d[1]$$

Сумма квадратов ошибок - функция потерь

In[50]:=
```
S4 = Total[(Y - f4[X]) ^2]
```
Out[50]=

$$(195.278 - d[0] - d[1])^2 + \left(413.238 - \frac{d[0]}{2} - d[1]\right)^2 + \left(241.814 - \frac{d[0]}{3} - d[1]\right)^2 + \left(297.835 - \frac{d[0]}{4} - d[1]\right)^2 + \left(399.894 - \frac{d[0]}{5} - d[1]\right)^2 + \left(537.373 - \frac{d[0]}{6} - d[1]\right)^2 + \left(371.448 - \frac{d[0]}{7} - d[1]\right)^2 + \left(356.627 - \frac{d[0]}{8} - d[1]\right)^2 + \left(577.297 - \frac{d[0]}{9} - d[1]\right)^2 + \left(671.979 - \frac{d[0]}{10} - d[1]\right)^2 + \left(423.869 - \frac{d[0]}{11} - d[1]\right)^2 + \left(603.441 - \frac{d[0]}{12} - d[1]\right)^2 + \left(863.965 - \frac{d[0]}{13} - d[1]\right)^2 + \left(705.427 - \frac{d[0]}{14} - d[1]\right)^2$$

Так как модель f4 линейна по параметрам, можно стандартным образом найти оптимальную точку.

Градиент функции потерь по переменным { d[0],d[1] }

In[51]:=
```
grad = D[S4, {{d[0], d[1]}}] // Simplify
```
Out[51]=

$$\{3.15199\,(-736.774 + d[0] + 2.06318\,d[1]),\ 6.50312\,(-2048.09 + d[0] + 4.30562\,d[1])\}$$

Система нормальных уравнений для определения оптимального значения параметров

In[52]:=
```
Thread[grad == 0] // MatrixForm
```
Out[52]//MatrixForm=

$$\begin{pmatrix} 3.15199\,(-736.774 + d[0] + 2.06318\,d[1]) == 0 \\ 6.50312\,(-2048.09 + d[0] + 4.30562\,d[1]) == 0 \end{pmatrix}$$

Решение этой системы

In[53]:=
```
{{dopt[0], dopt[1]}} = SolveValues[Thread[grad == 0], {d[0], d[1]}]
```
Out[53]=

$$\{\{-469.712,\ 584.77\}\}$$

Оптимальная гиперболическая модель

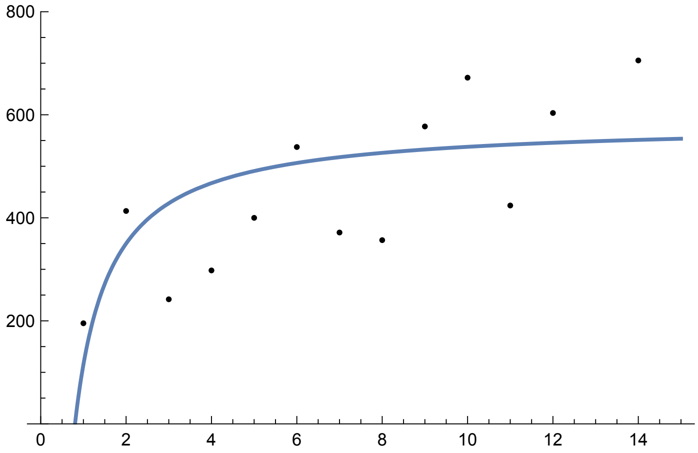
In[54]:=
```
f4opt[x_] = dopt[1] + dopt[0] / x
```
Out[54]=

$$584.77 - \frac{469.712}{x}$$

Совместный график исходных данных и построенной оптимальной функции

In[55]:=
```
Plot[f4opt[x], {x, 0, 15}, PlotRange → {0, 800}, Epilog → Point[data]]
```

Out[55]=



Вычислим критерии качества - коэффициент детерминации R^2 и MAPE.

In[56]:=
```
SSE4 = Total[(Y - f4opt[X])^2]
```

Out[56]=
```
293 756.
```

In[57]:=
```
SST4 = Total[(Y - Mean[Y])^2]
```

Out[57]=
```
474 849.
```

In[58]:=
```
r2[4] = 1 - SSE4 / SST4
r2[4] = Round[r2[4], 0.001]
```

Out[58]=
```
0.381371
```

Out[59]=
```
0.381
```

In[60]:=
```
MAPE[4] = Mean[Abs[Y - f4opt[X]] / Abs[Y]]
MAPE[4] = Round[MAPE[4], 0.001]
```

Out[60]=
```
0.306639
```

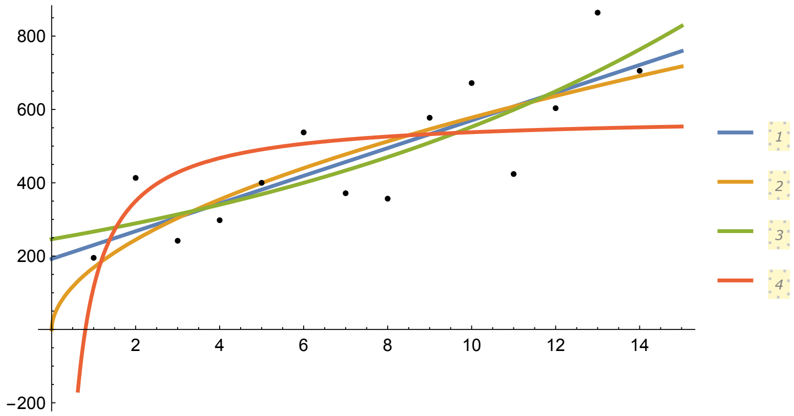
Out[61]=
```
0.307
```

3. Сравнение моделей

Графики всех построенных моделей

In[62]:=
```
Plot[{f1opt[x], f2opt[x], f3opt[x], f4opt[x]},
 {x, 0, 15}, Epilog → {Point[data]}, PlotLegends → Automatic]
```

Out[62]=



Сравнение качества моделей проводится по двум критериям - коэффициенту детерминации R^2 и средней относительной ошибки в процентах (MAPE).

In[63]:=
```
Grid[{{"Модель", "Формула", "R²", "MAPE"},
  {"Линейная", f1opt[x], r2[1], MAPE[1]},
  {"Степенная", f2opt[x], r2[2], MAPE[2]},
  {"Показательная", f3opt[x], r2[3], MAPE[3]},
  {"Гиперболическая", f4opt[x], r2[4], MAPE[4]}}, Frame → All]
```

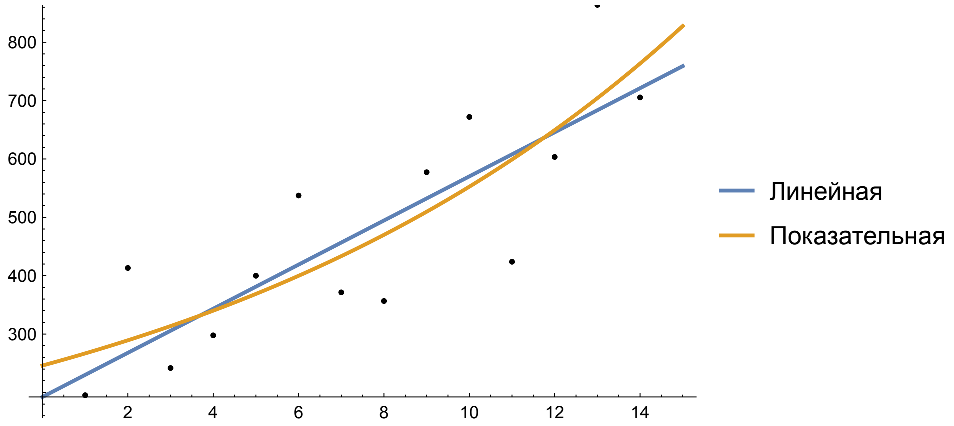
Out[63]=

Модель	Формула	R^2	MAPE
Линейная	$192.29 + 37.785\, x$	0.684	0.2
Степенная	$169.067\, x^{0.533644}$	0.65	0.202
Показательная	$246.121 \times 1.08422^{x}$	0.696	0.213
Гиперболическая	$584.77 - \frac{469.712}{x}$	0.381	0.307

Гиперболическая модель проигрывает всем остальным моделям по обоим критериям (самое маленькое значение R^2 и свмое большое значение MAPE.
Первые три модели имеют практически однаковые значения MAPE, но лучшей является линейная модель, а по критерию R^2 лучшей является показательная модель, с незначительным опережением линейной модели.
Таким образом выбор среди двух моделей, линейной и показательной.
Линейная модель может быть выбрана в качестве основной, так как она, дополнительно, проще устроена (критерий простоты модели).

In[64]:=
```
Plot[{f1opt[x], f3opt[x]}, {x, 0, 15},
 Epilog → {Point[data]}, PlotLegends → {"Линейная", "Показательная"}]
```

Out[64]=



4. Использование стандартных средств WM

In[65]:=
```
data
```

Out[65]=
```
{{1, 195.278}, {2, 413.238}, {3, 241.814}, {4, 297.835}, {5, 399.894},
 {6, 537.373}, {7, 371.448}, {8, 356.627}, {9, 577.297}, {10, 671.979},
 {11, 423.869}, {12, 603.441}, {13, 863.965}, {14, 705.427}}
```

Стандартная функция FindFit [данные, модель, параметры, переменная].
Для заданной модели (формула для модели) определяются наилучшие значения параметров. Для линейных по параметрам моделей (модели 1 и 4) используется Метод наименьших квадратов.

In[66]:=
```
g1[x_] = (a + b x) /. FindFit[data, a + b x, {a, b}, x]
g2[x_] = (a x ^ b) /. FindFit[data, a x ^ b, {a, b}, x]
g3[x_] = (a b ^ x) /. FindFit[data, a b ^ x, {a, b}, x]
g4[x_] = (a + b / x) /. FindFit[data, a + b / x, {a, b}, x]
```

Out[66]=

$192.29 + 37.785\, x$

Out[67]=

$169.067\, x^{0.533644}$

Out[68]=

$246.121 \times 1.08422^{x}$

Out[69]=

$$584.77 - \frac{469.712}{x}$$

In[70]:=
```
{f1opt[x], f2opt[x], f3opt[x], f4opt[x]}
```

Out[70]=

$$\left\{192.29 + 37.785\, x,\ 169.067\, x^{0.533644},\ 246.121 \times 1.08422^{x},\ 584.77 - \frac{469.712}{x}\right\}$$

Полученные функции совпадают с полученными ранее зависимостями. Это означает, что стандартные функции, по умолчанию, используют метод наименьших квалратов.