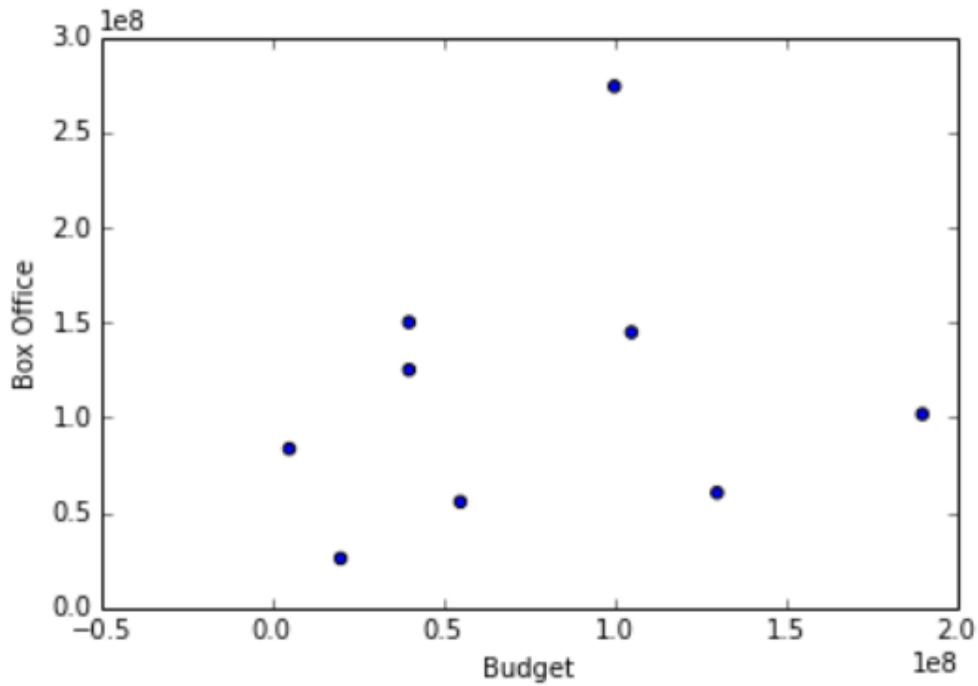
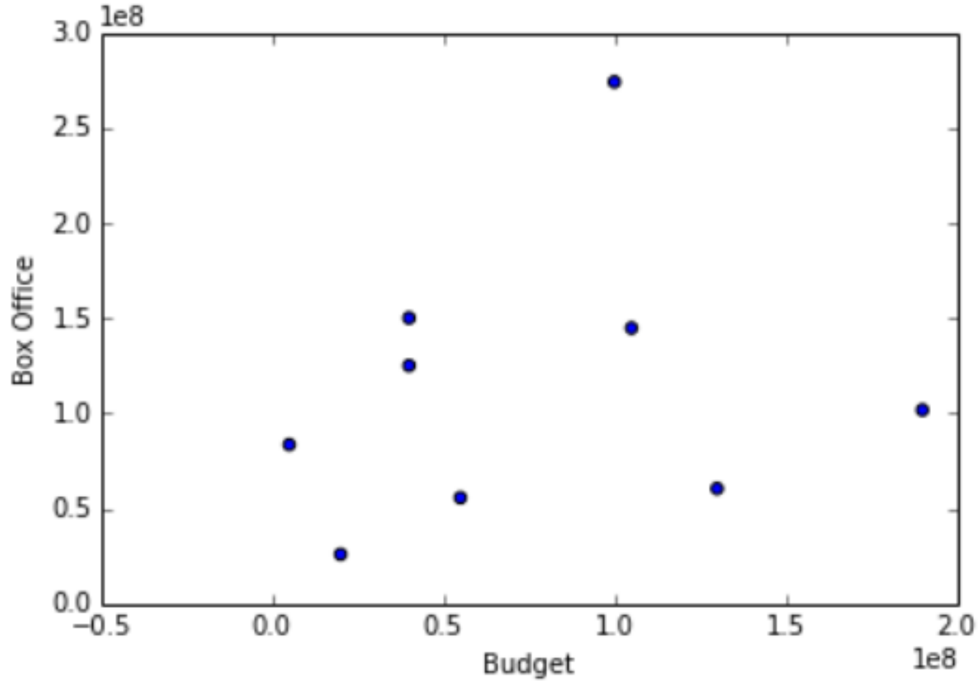
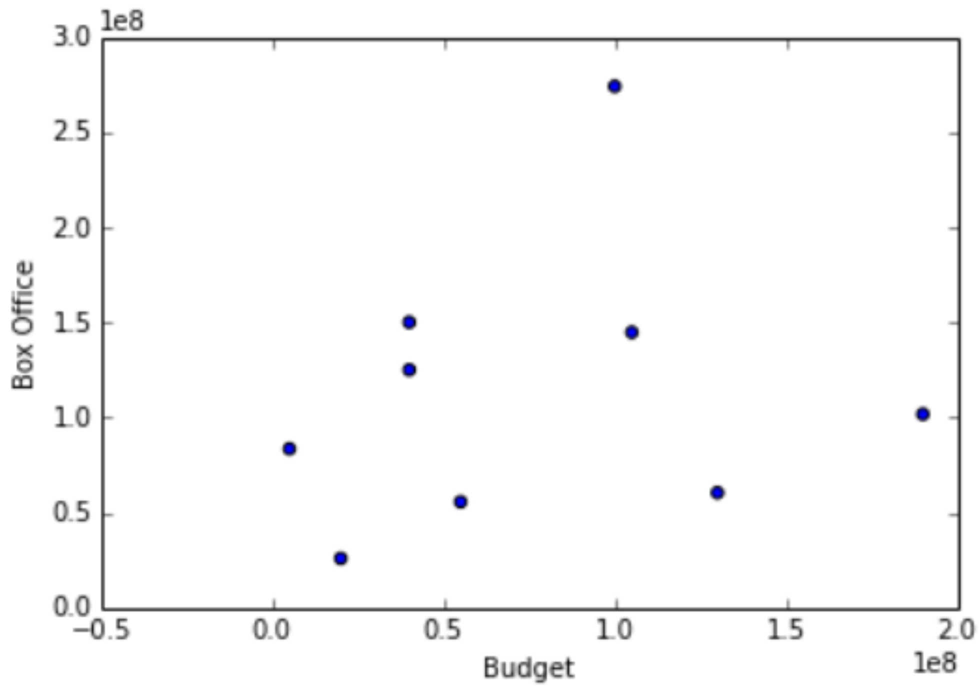
Linear Regression


1e8
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Box Office
−0.5
0.0
0.5
1.0
1.5
2.0
Budget
1e8



$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$



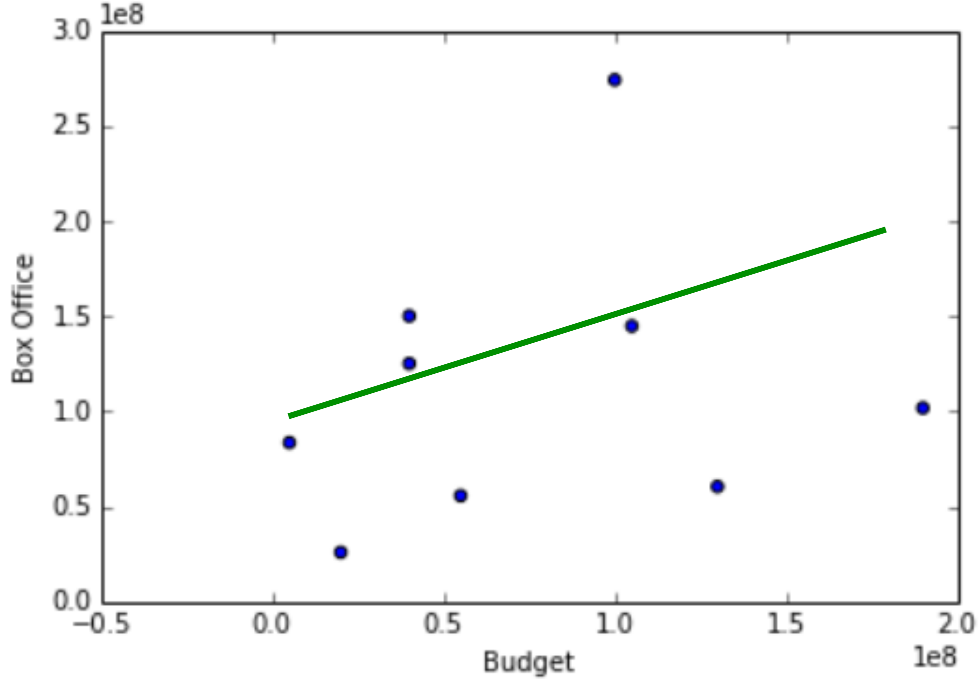
coef 0 coef 1

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

Gross of movie

Budget of movie

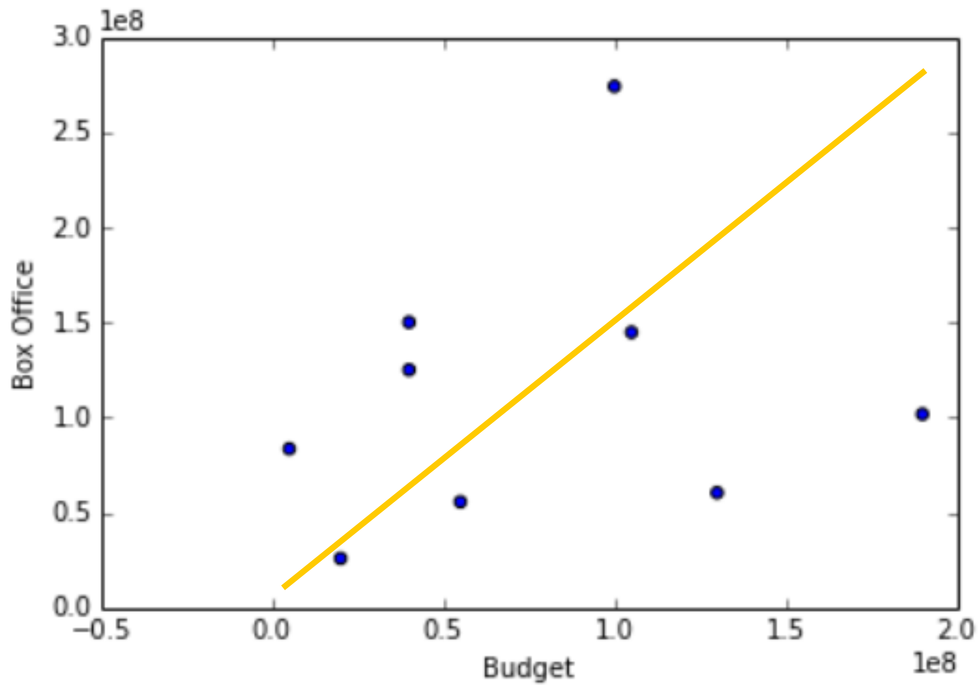
Noise (random for each movie)



$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

$\beta_0 = 80million$

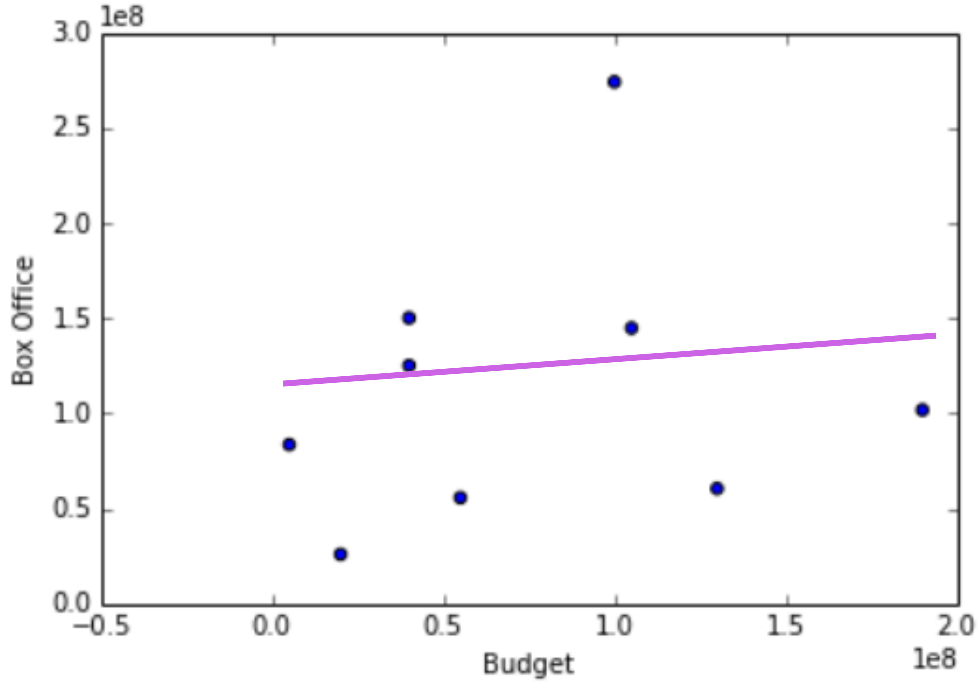
$\beta_1 = 0.5$



$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

$\beta_0 = 80 million$
$\beta_1 = 0.5$

$\beta_0 = 0$
$\beta_1 = 1.5$

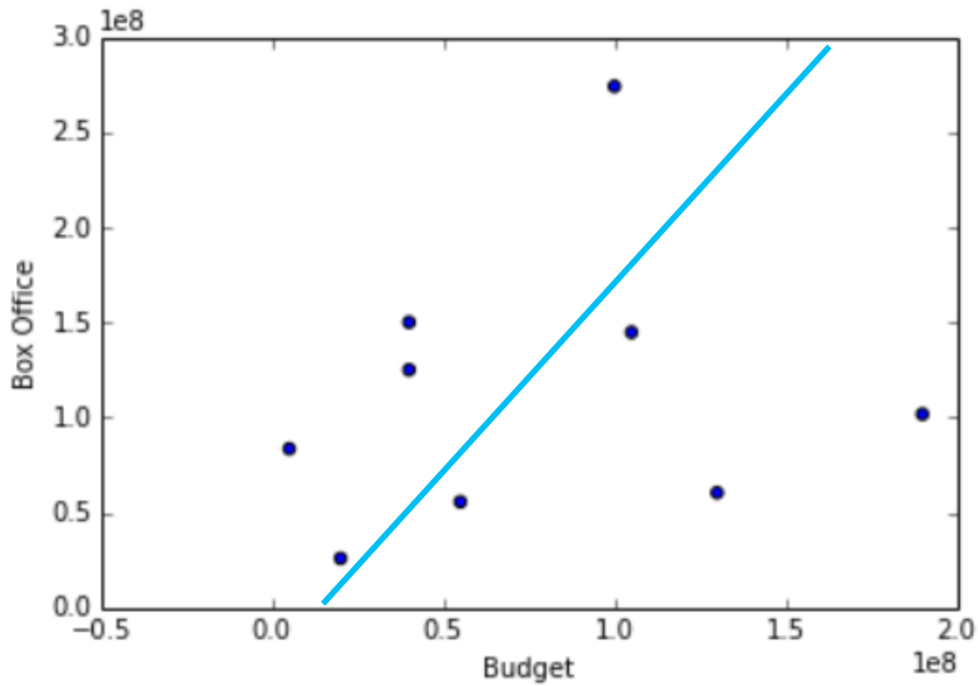


$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

$\beta_0 = 80 million$
$\beta_1 = 0.5$

$\beta_0 = 0$
$\beta_1 = 1.5$

$\beta_0 = 120 million$
$\beta_1 = 0.1$



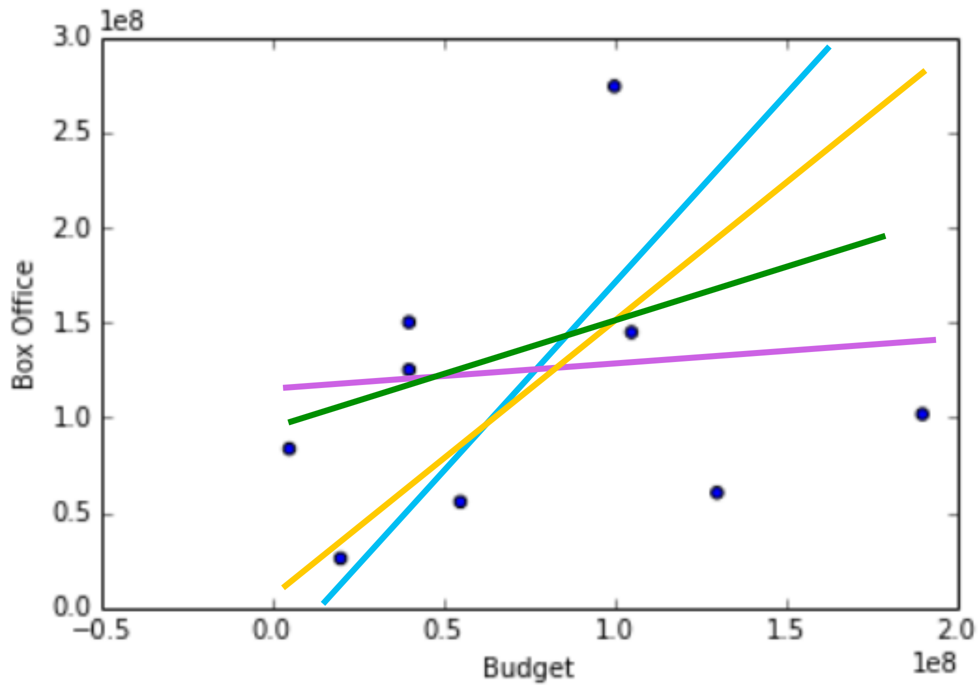
$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

$\beta_0 = 80million$
$\beta_1 = 0.5$

$\beta_0 = 0$
$\beta_1 = 1.5$

$\beta_0 = 120million$
$\beta_1 = 0.1$

$\beta_0 = 30million$
$\beta_1 = 2$



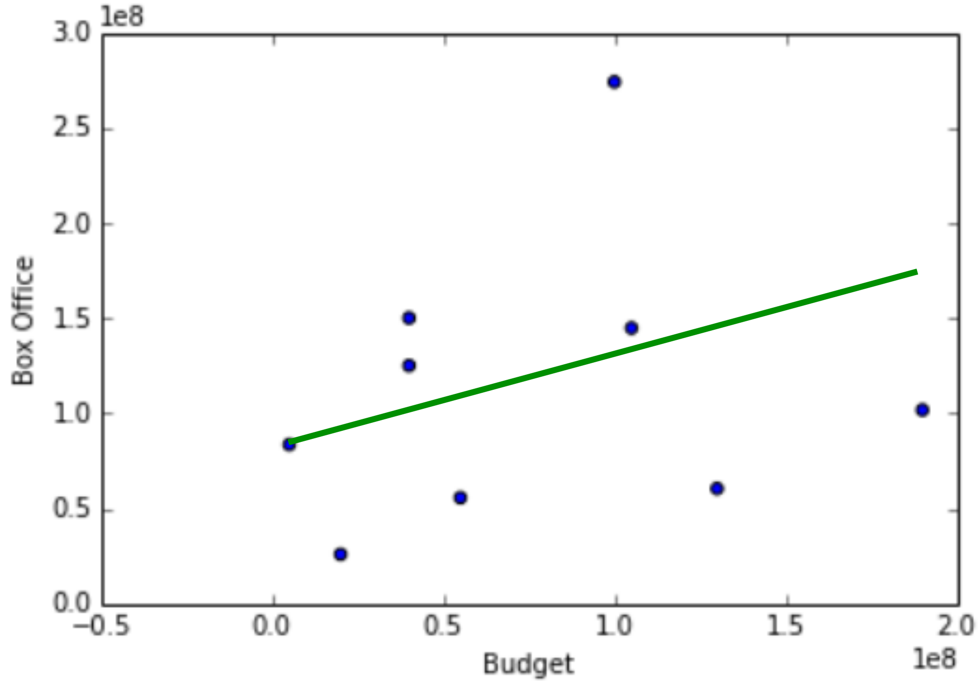
$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

$\beta_0 = 80 million$
$\beta_1 = 0.5$

$\beta_0 = 0$
$\beta_1 = 1.5$

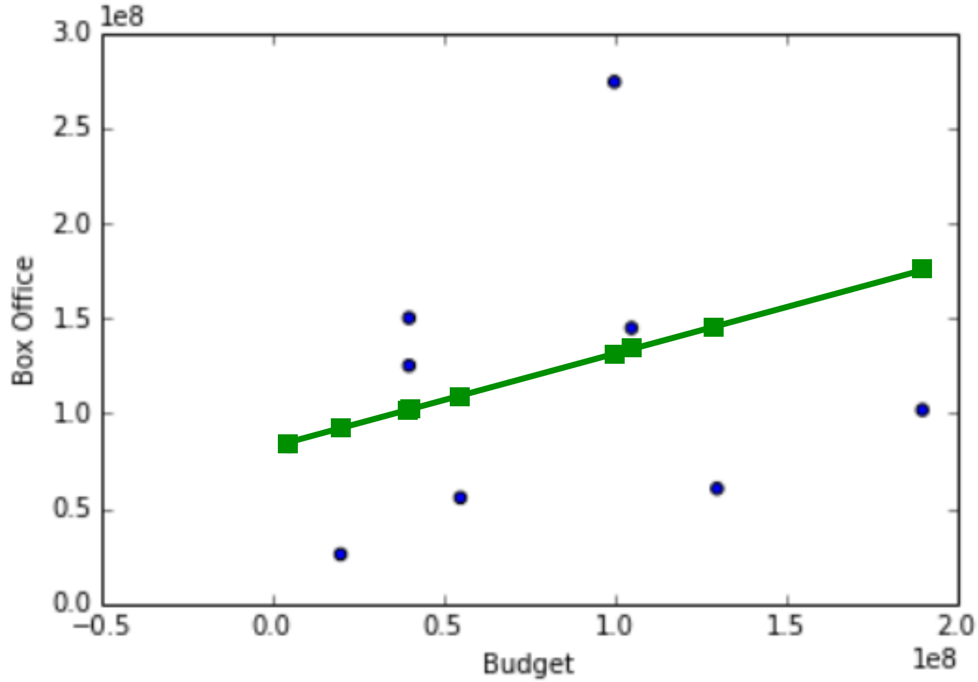
$\beta_0 = 120 million$
$\beta_1 = 0.1$

$\beta_0 = 30 million$
$\beta_1 = 2$



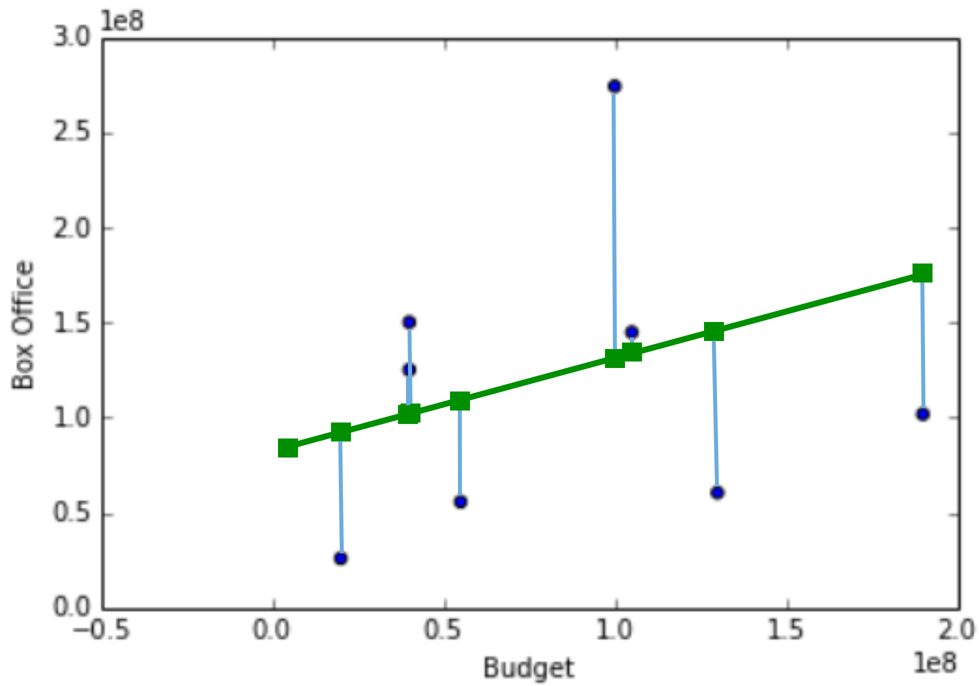
$$y_{\beta}(x) = \beta_0 + \beta_1 x + \varepsilon$$

$\beta_0 = 80million$	$\beta_0 = 0$	$\beta_0 = 120million$	$\beta_0 = 30million$
$\beta_1 = 0.5$	$\beta_1 = 1.5$	$\beta_1 = 0.1$	$\beta_1 = 2$



$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

$\beta_0 = 80million$	$\beta_0 = 0$	$\beta_0 = 120million$	$\beta_0 = 30million$
$\beta_1 = 0.5$	$\beta_1 = 1.5$	$\beta_1 = 0.1$	$\beta_1 = 2$



$$y_\beta(x_{obs}^{(0)}) - y_{obs}^{(0)}$$

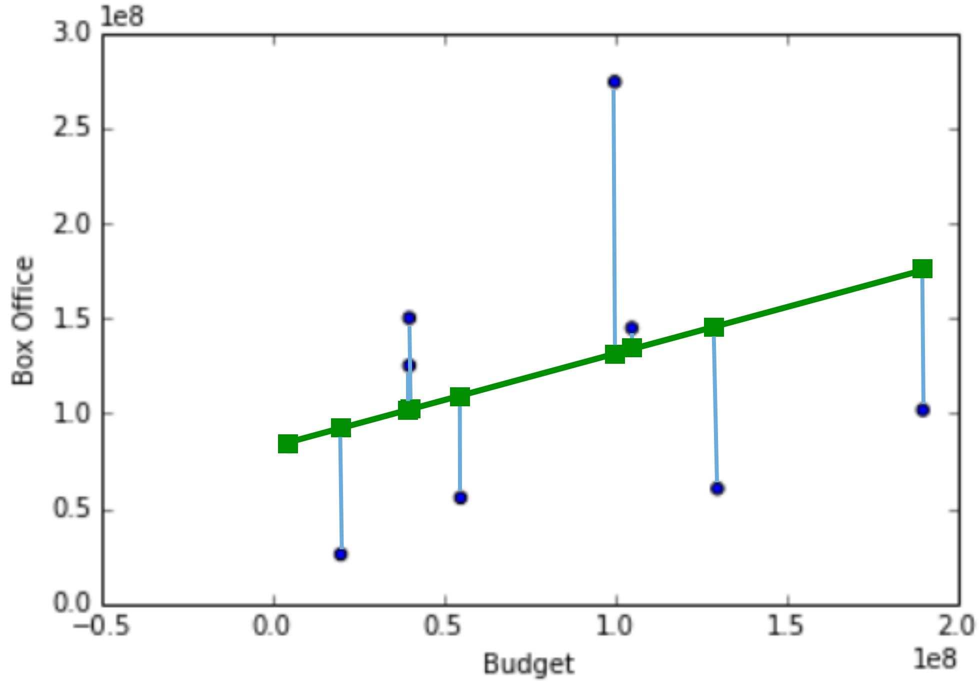
$$y_\beta(x_{obs}^{(1)}) - y_{obs}^{(1)}$$

$$y_\beta(x_{obs}^{(2)}) - y_{obs}^{(2)}$$

$$y_\beta(x_{obs}^{(3)}) - y_{obs}^{(3)}$$

Predicted value by model – Observed value

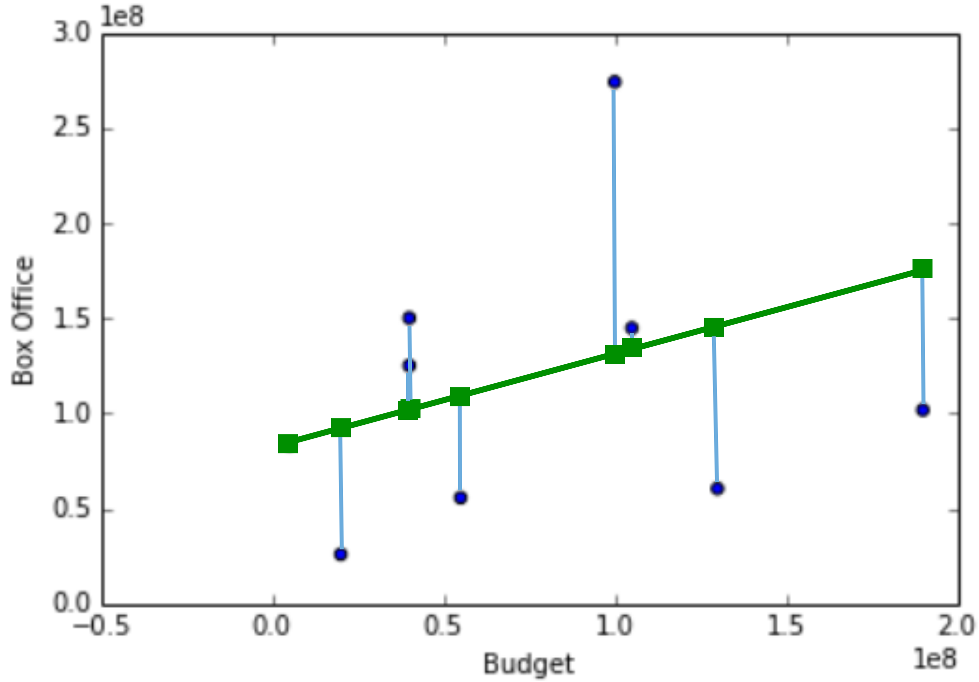
β0 =80M, β1=0.5



Predicted value by model – Observed value

β0 =80M, β1=0.5

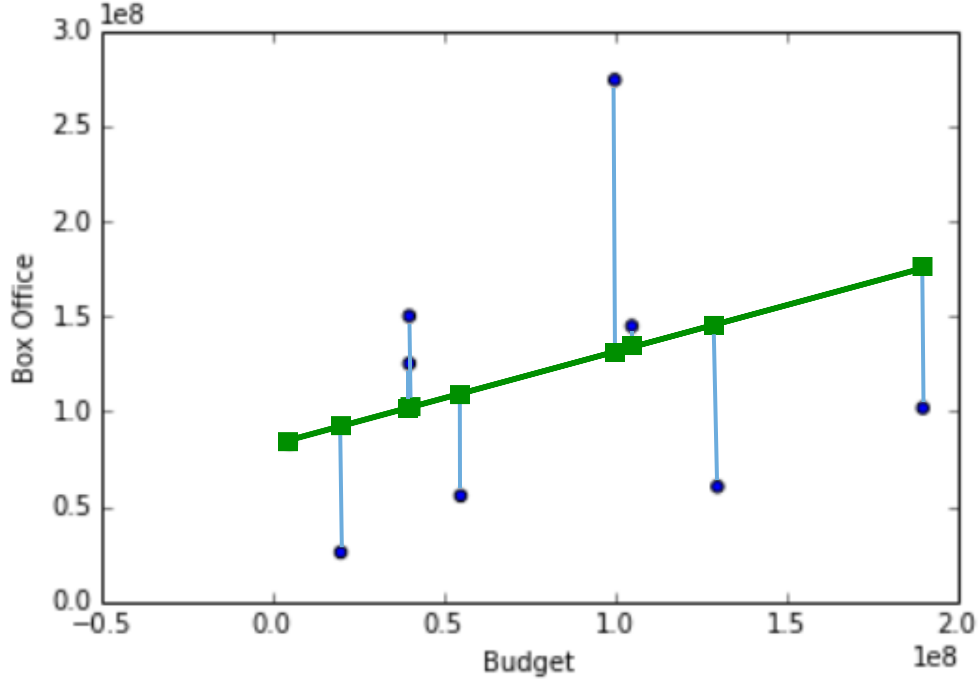
$$y_{\beta}(x_{obs}^{(i)}) - y_{obs}^{(i)}$$



Predicted value by model – Observed value

β0 =80M, β1=0.5

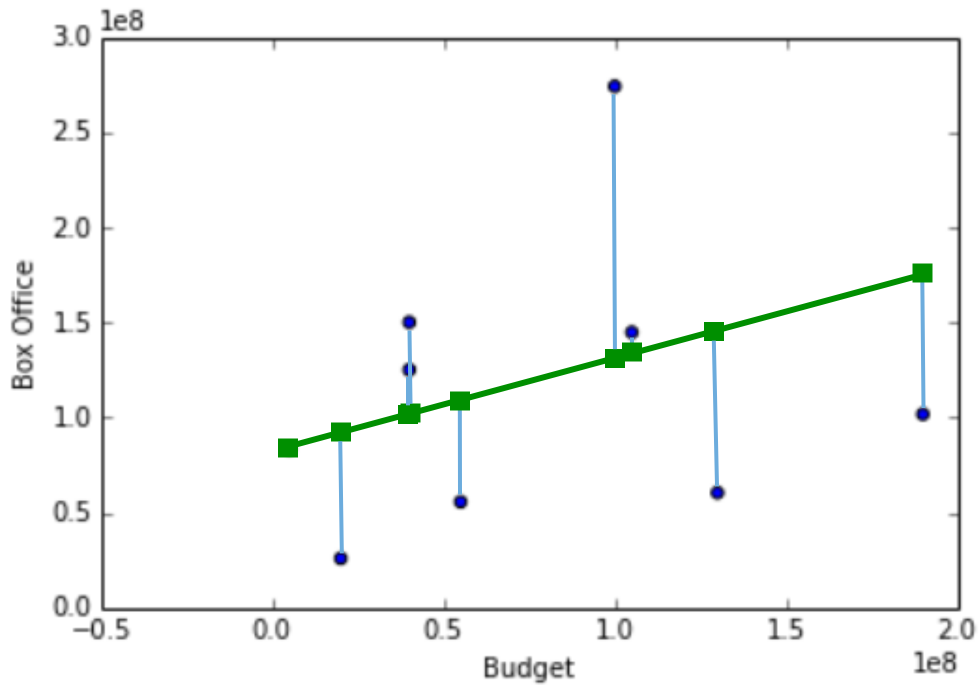
$$(\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)}$$



Predicted value by model – Observed value

β0 =80M, β1=0.5

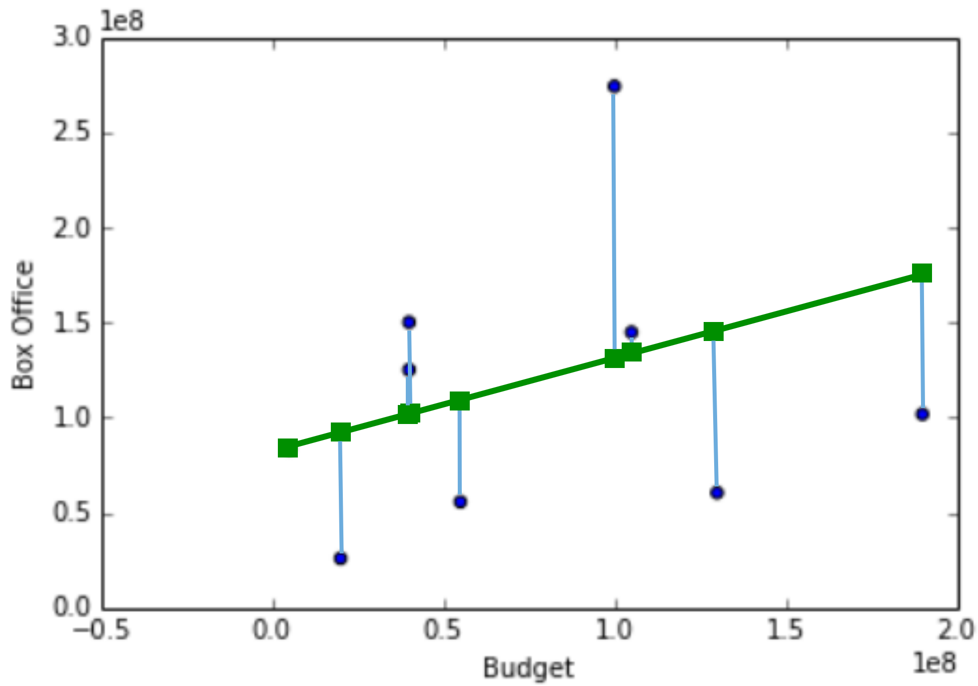
$$\sum_{i=1}^{m}\left((\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)}\right)^2$$



Predicted value by model – Observed value

β0 =80M, β1=0.5

$$\min_{\beta_0,\beta_1} \sum_{i=1}^{m} \left((\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)} \right)^2$$

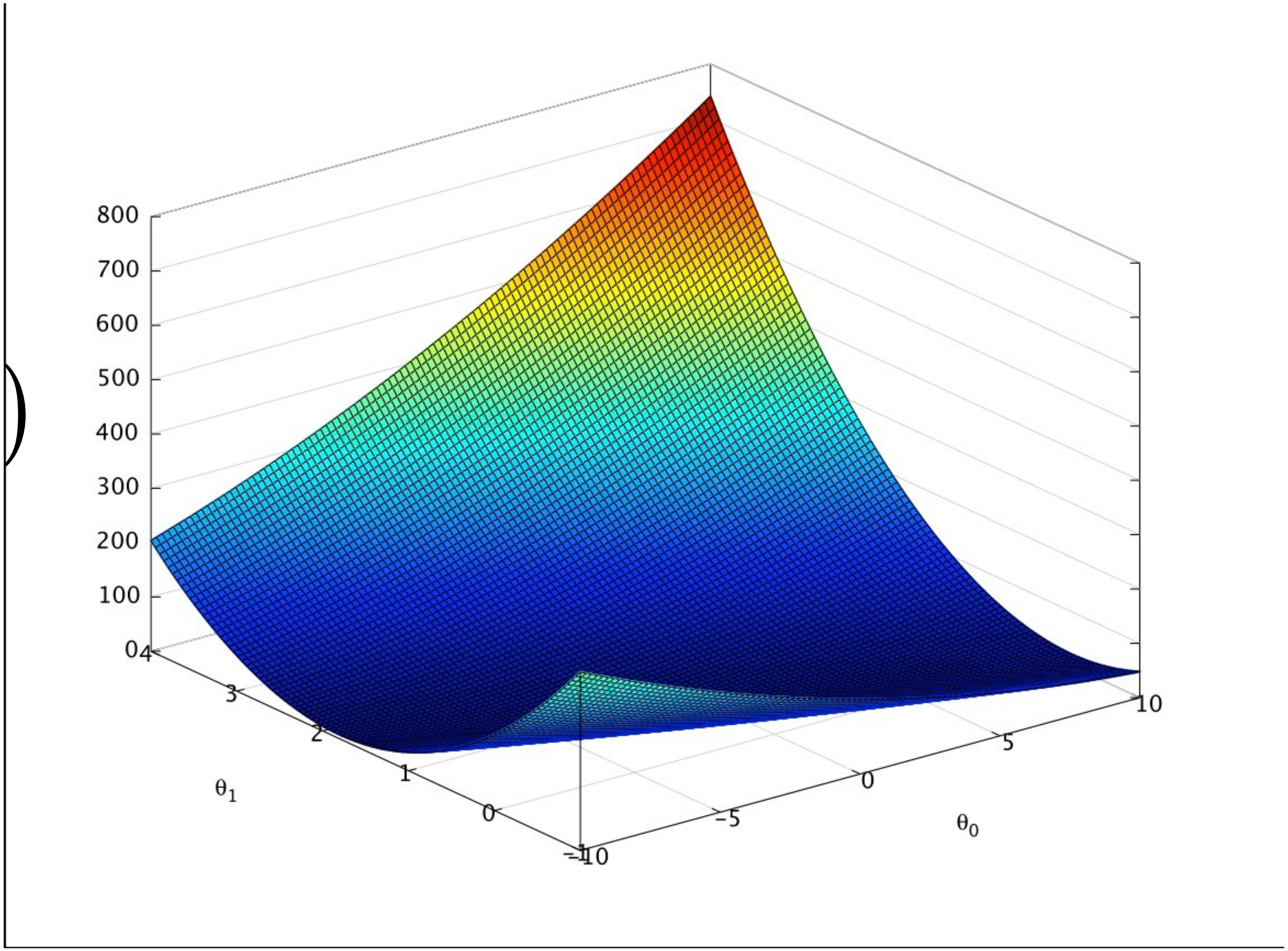


Cost function

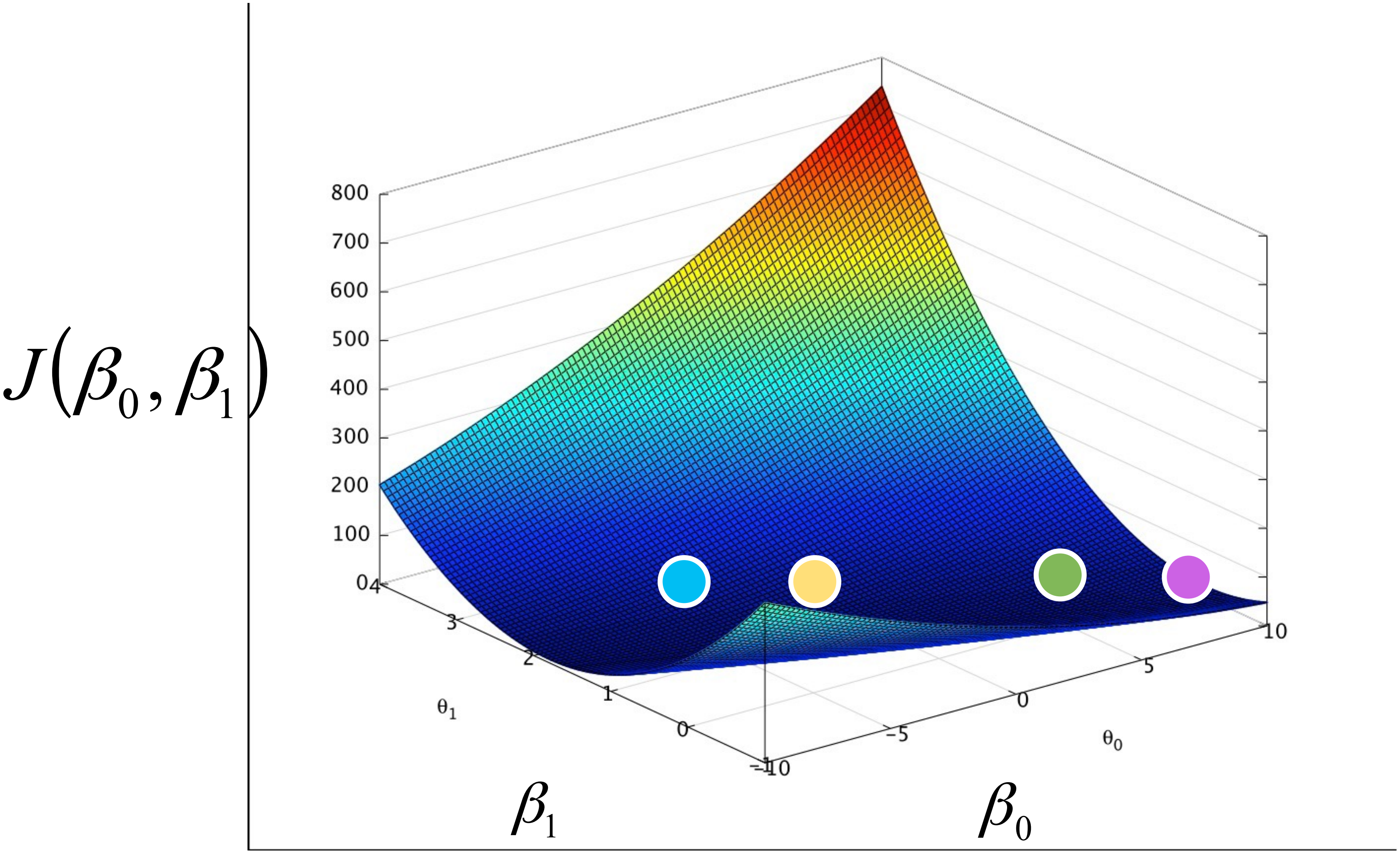
Takes a model (specific parameter values), returns score

$$J(\beta_0, \beta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left((\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)} \right)^2$$

$J(\beta_0, \beta_1)$



$$J(\beta_0, \beta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left((\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)} \right)^2$$

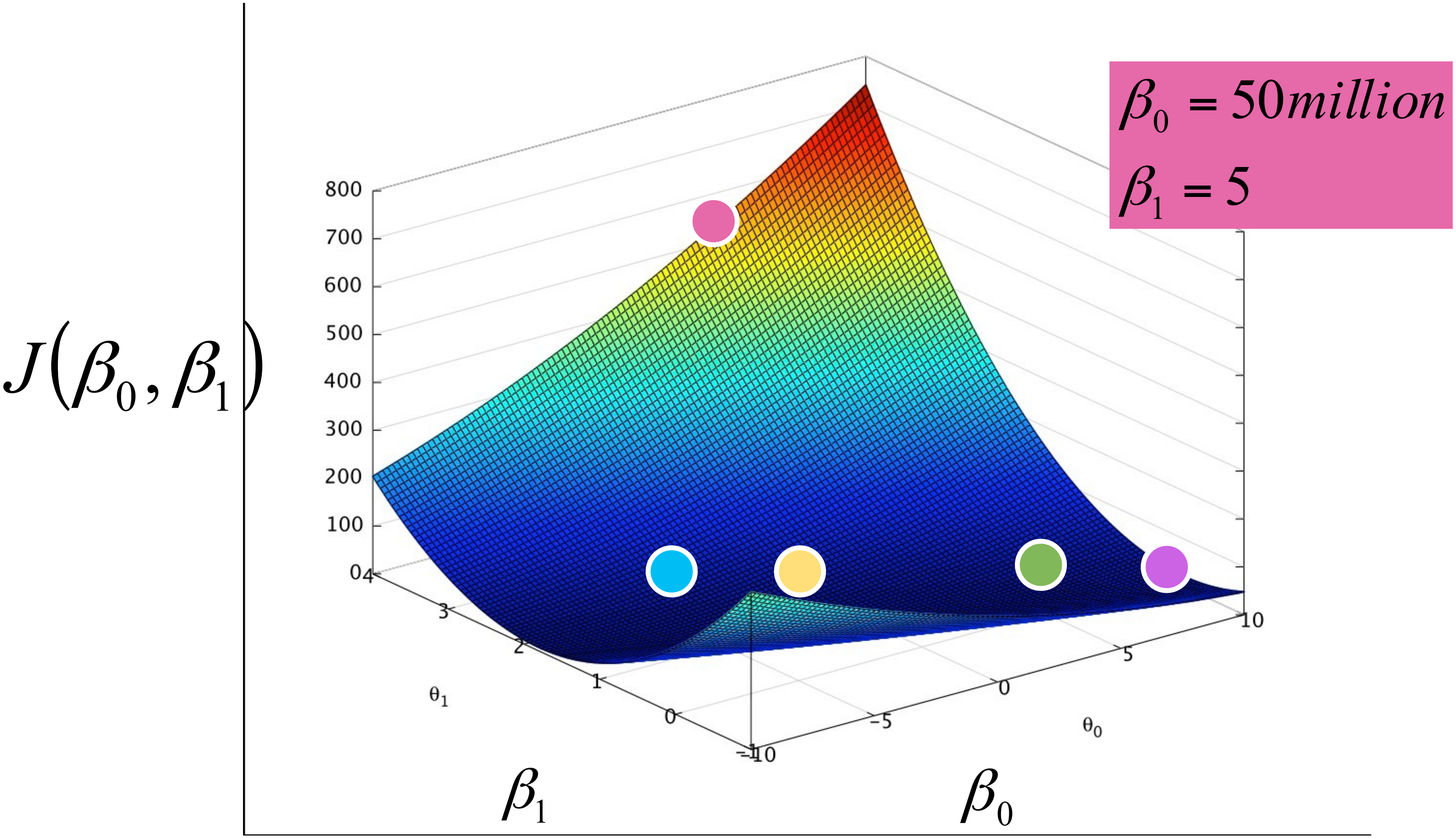


$\beta_0 = 80 million$
$\beta_1 = 0.5$

$\beta_0 = 0$
$\beta_1 = 1.5$

$\beta_0 = 120 million$
$\beta_1 = 0.1$

$\beta_0 = 30 million$
$\beta_1 = 2$



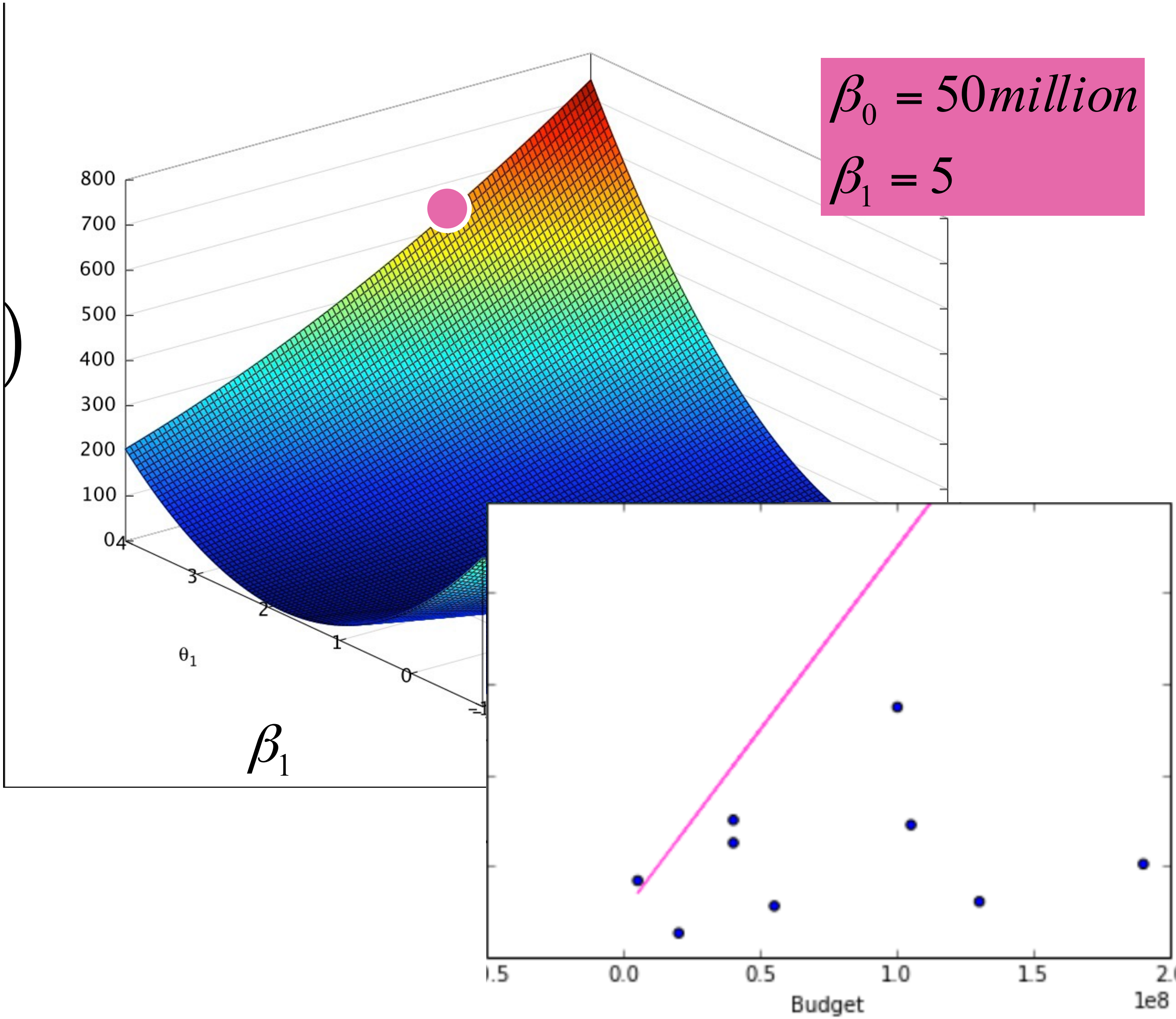
$\beta_0 = 80million$
$\beta_1 = 0.5$

$\beta_0 = 0$
$\beta_1 = 1.5$

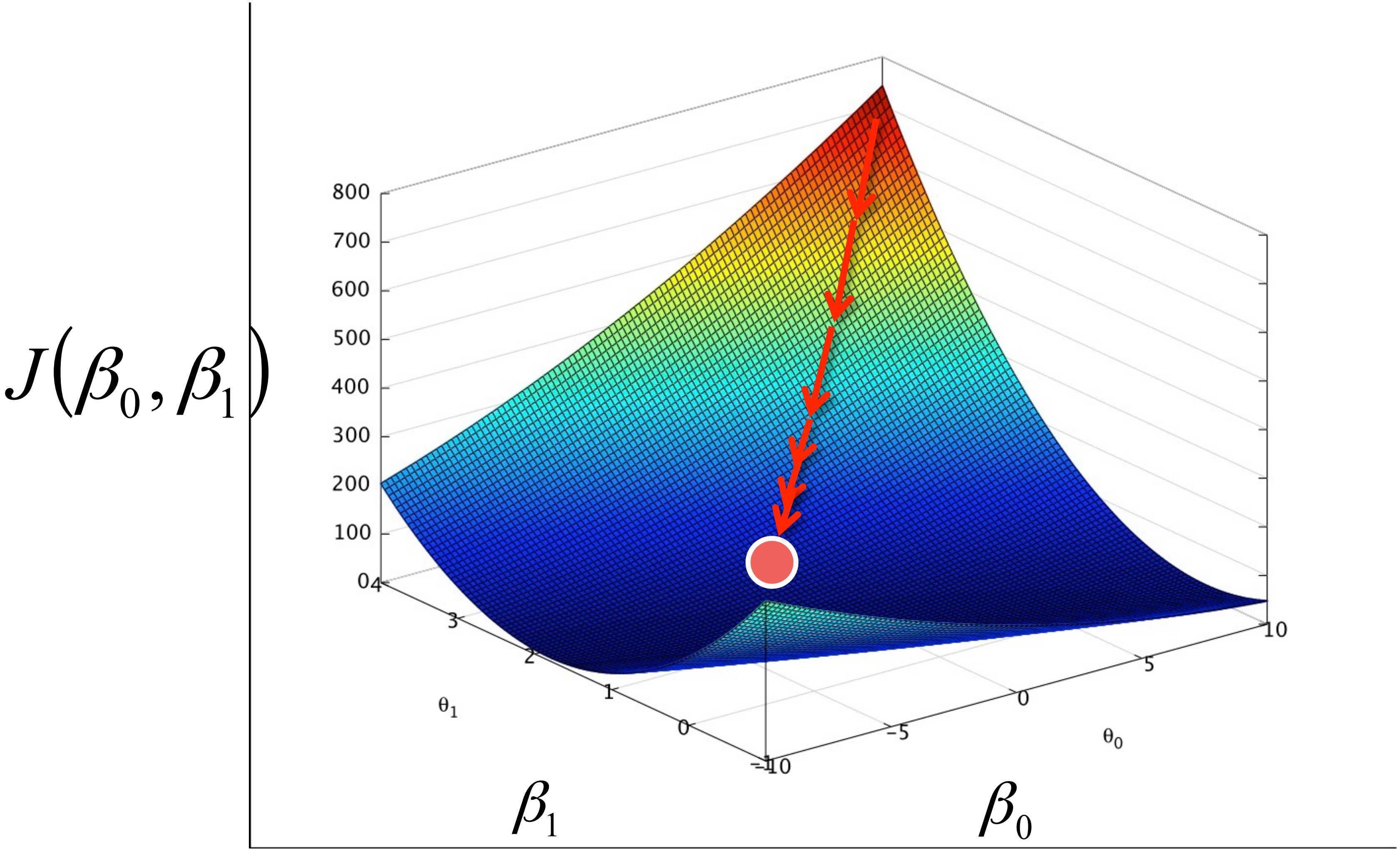
$\beta_0 = 120million$
$\beta_1 = 0.1$

$\beta_0 = 30million$
$\beta_1 = 2$

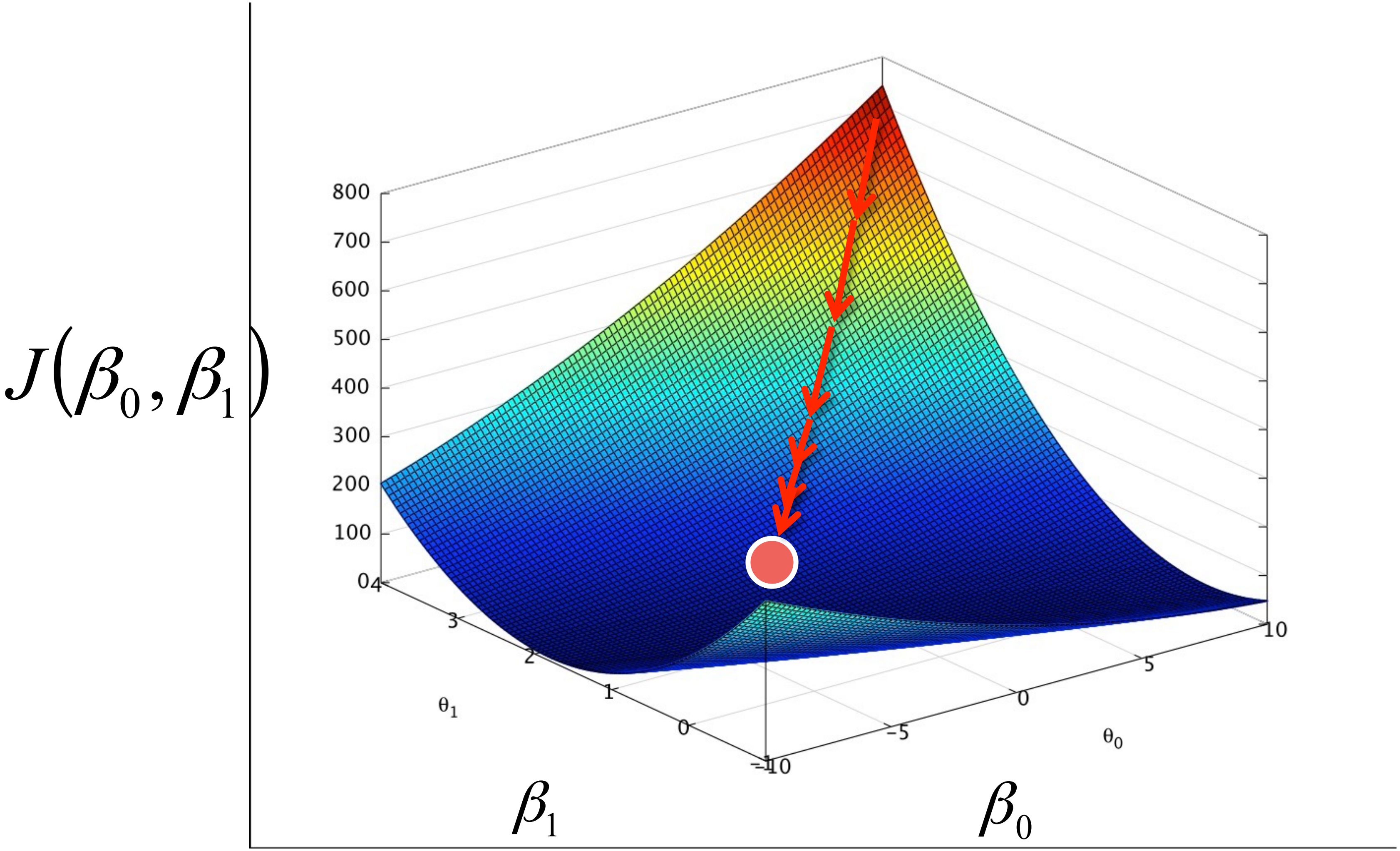
$$J(\beta_0, \beta_1)$$



$$\beta_0 = 50million$$
$$\beta_1 = 5$$

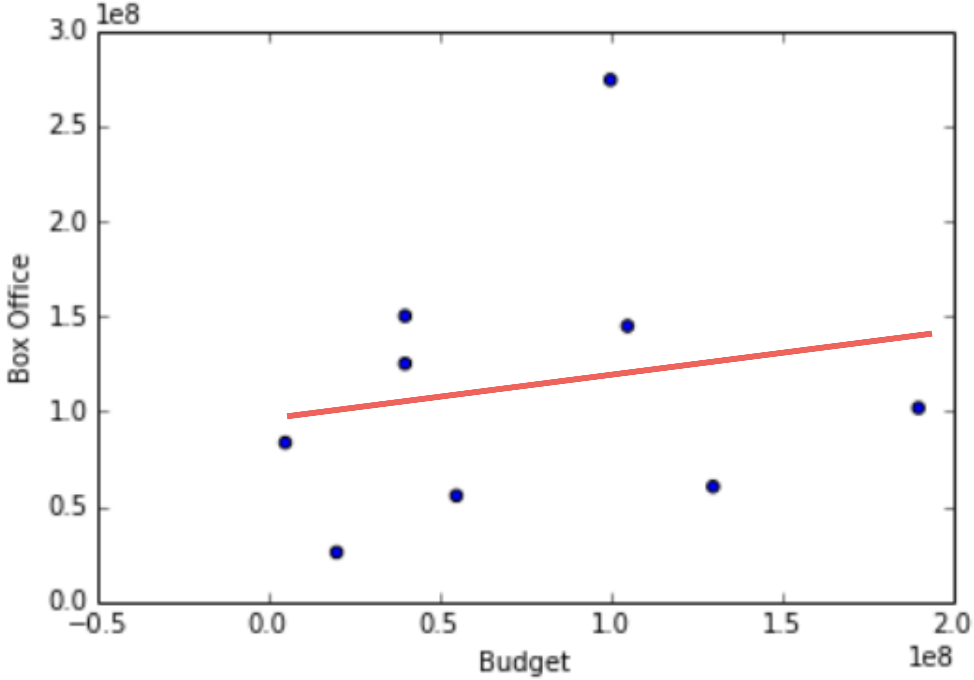



```
import statsmodels.formula.api as sm
linmodel = sm.OLS(Y, X).fit()
```



$$\beta_0 = 94.68 million$$

$$\beta_1 = 0.1$$

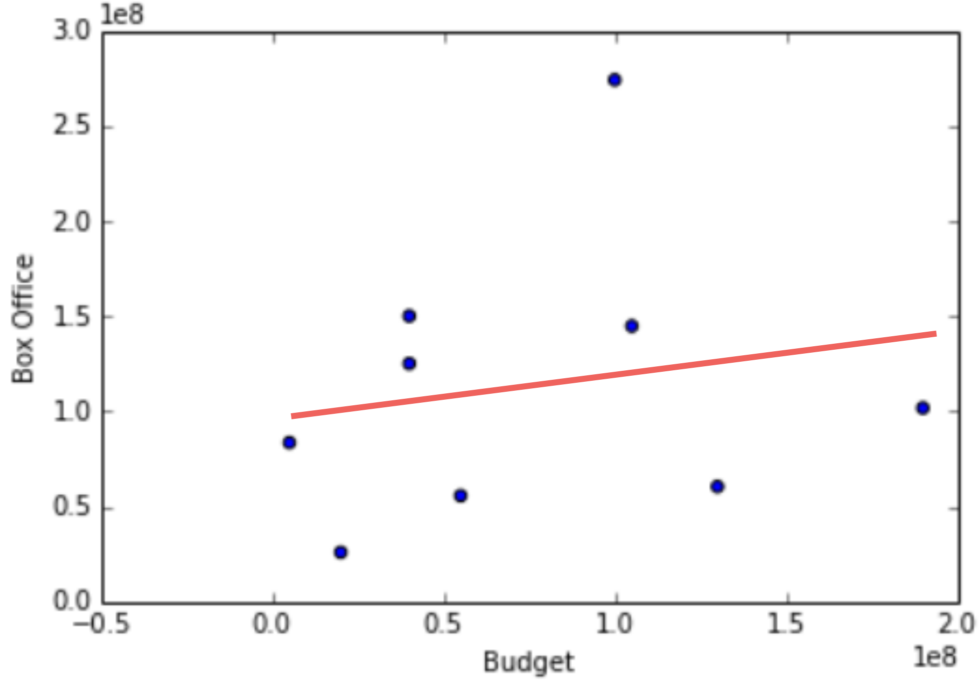


$$y_{\beta}(x) = \beta_0 + \beta_1 x + \varepsilon$$

$$\beta_0 = 94.68 million$$
$$\beta_1 = 0.1$$

Models and Randomness

DATA SCIENCE BOOTCAMP

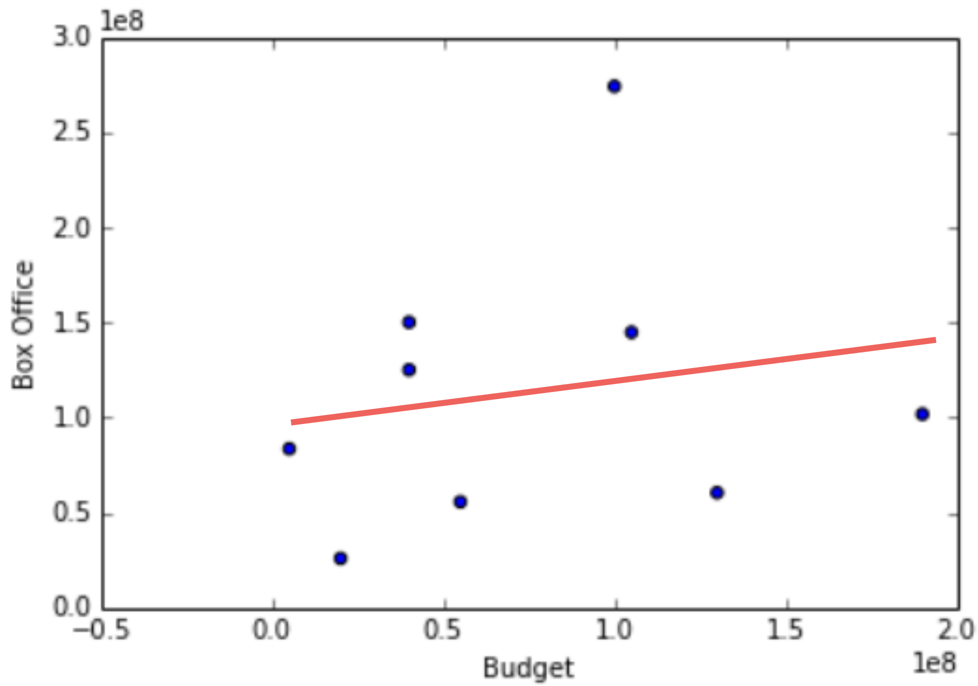


$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

Random for each movie

$$\beta_0 = 94.68 million$$

$$\beta_1 = 0.1$$



$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

Random
Normal
distribution
Mean=0
Stdev=
$67,762,000

$$\beta_0 = 94.68 million$$
$$\beta_1 = 0.1$$

$$\beta_0 = 94.68 million$$
$$\beta_1 = 0.1$$

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

Random
Normal distribution
Mean=0
Stdev= $67,762,000

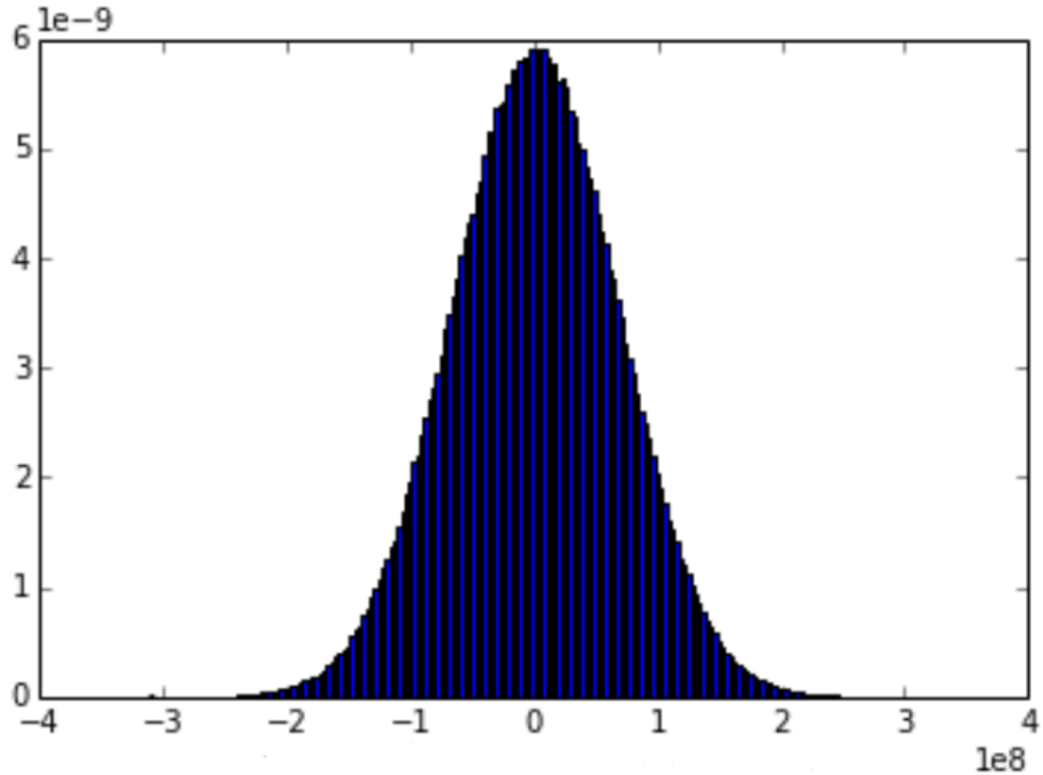
$\beta_0 = 94.68 million$

$\beta_1 = 0.1$

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

Random
Normal distribution
Mean=0
Stdev= \$67,762,000

pdf(ε)



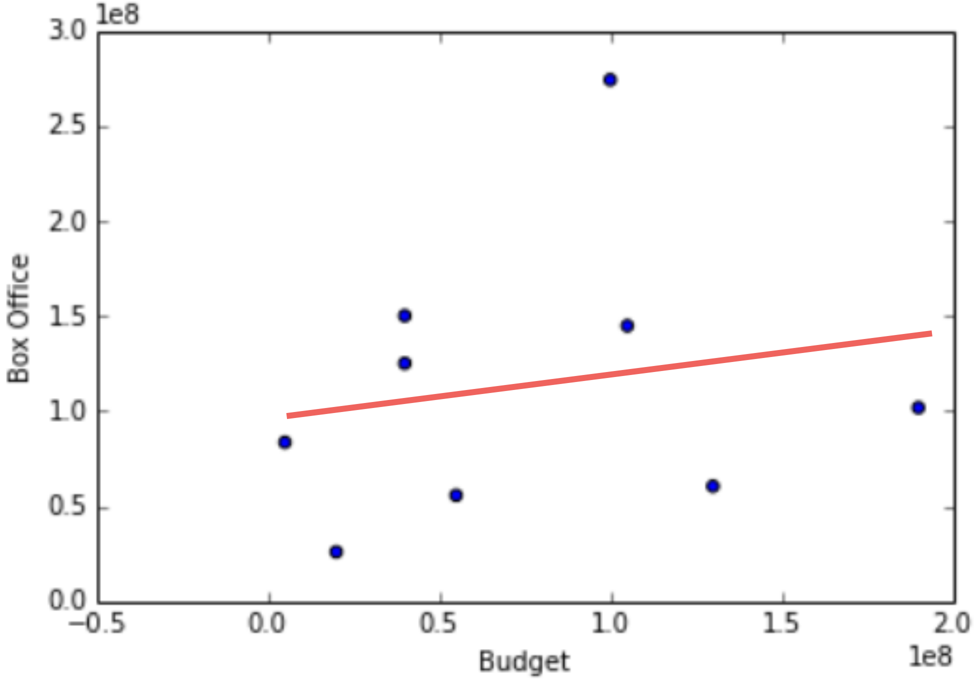
ε

(Note axes!)

$$\beta_0 = 94.68 million$$
$$\beta_1 = 0.1$$

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

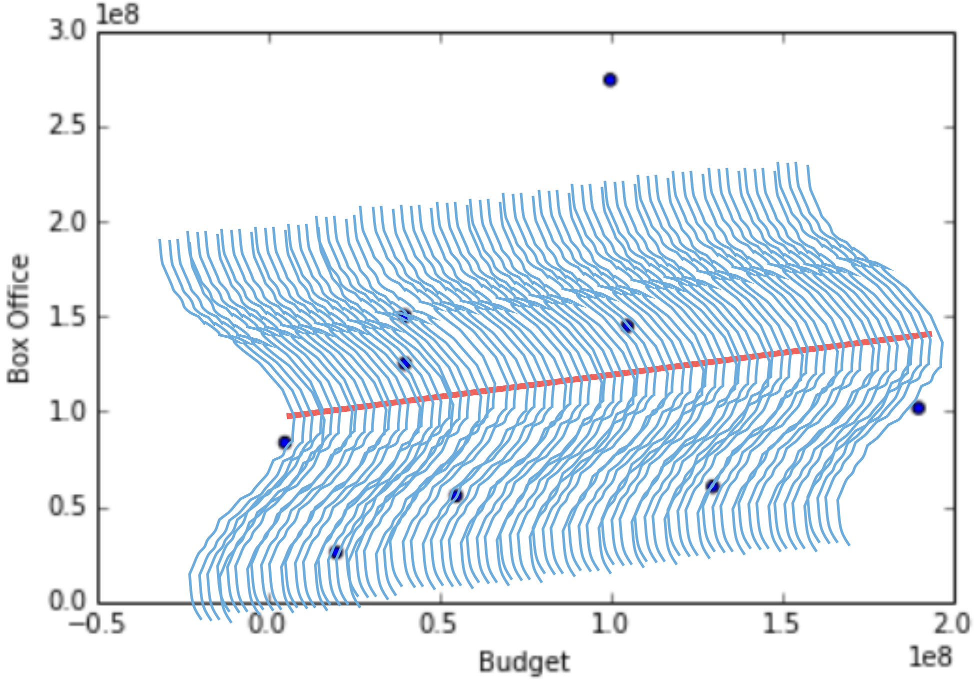
Random
Normal distribution
Mean=0
Stdev= $67,762,000



$$\beta_0 = 94.68 million$$
$$\beta_1 = 0.1$$

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

Random
Normal distribution
Mean=0
Stdev= $67,762,000



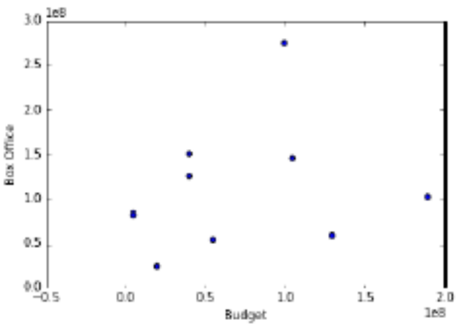
$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

```
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

```
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```

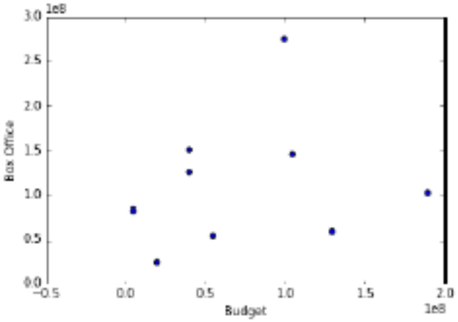
Our world
(observed)



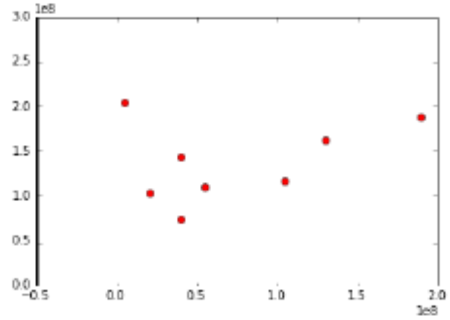
$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

```
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```

Our world (observed)



Alternate universe 1

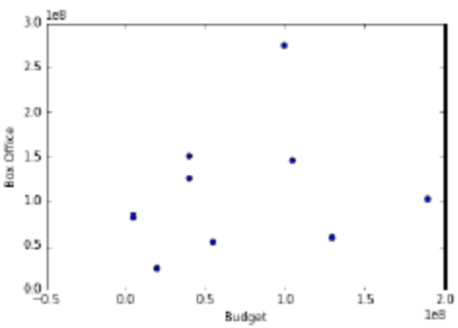


Same budgets
Same model
Different grosses

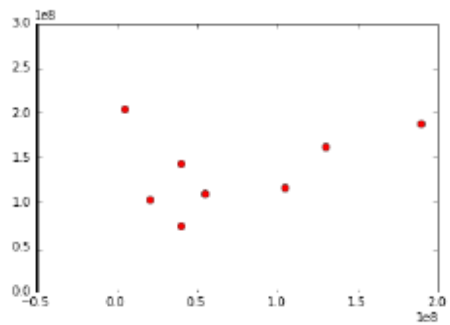
$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

```
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```

Our world (observed)

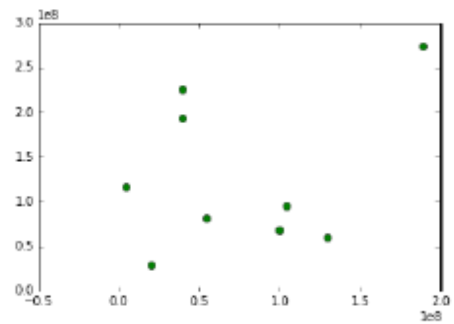


Alternate universe 1



Same budgets
Same model
Different grosses

Alternate universe 2

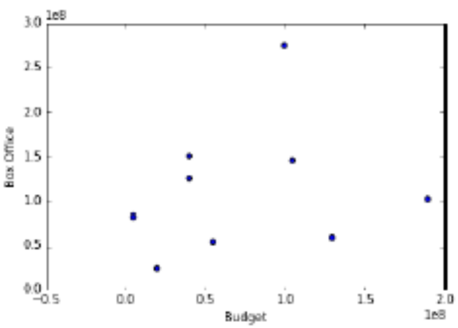


Same budgets
Same model
Different grosses

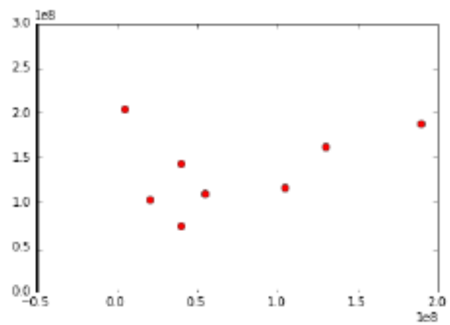
$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

```python
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```

Our world (observed)

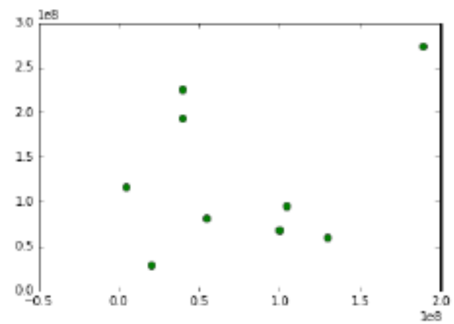


Alternate universe 1



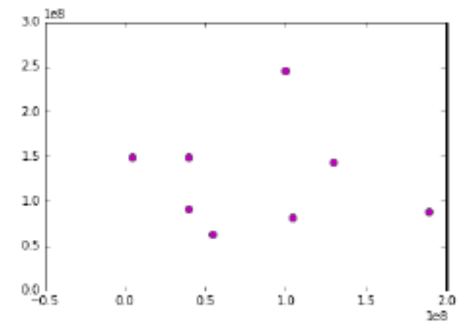
Same budgets
Same model
Different grosses

Alternate universe 2



Same budgets
Same model
Different grosses

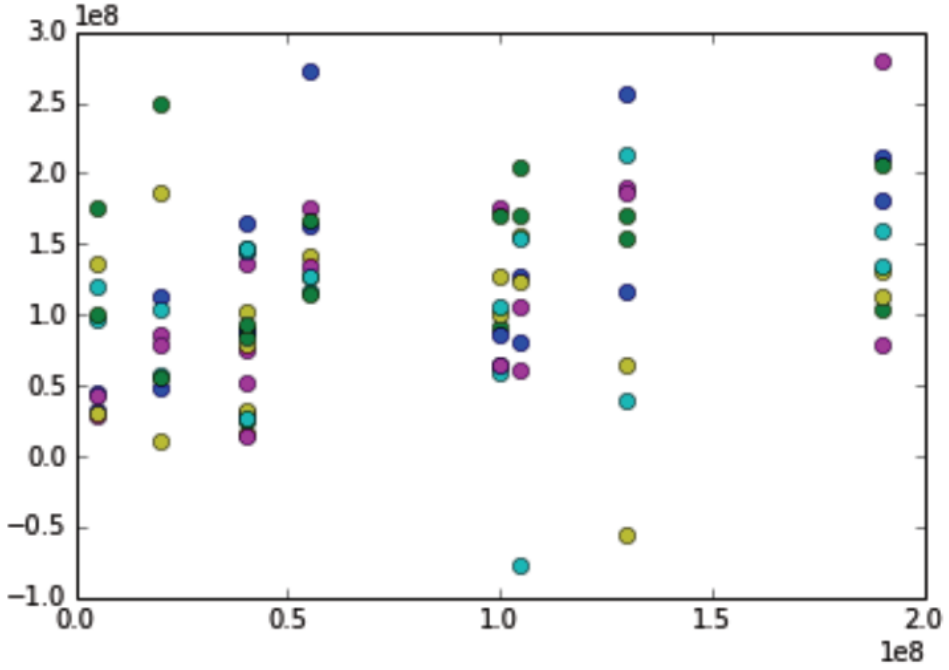
Alternate universe 3



Same budgets
Same model
Different grosses

$$y_{\beta}(x) = \beta_0 + \beta_1 x + \varepsilon$$

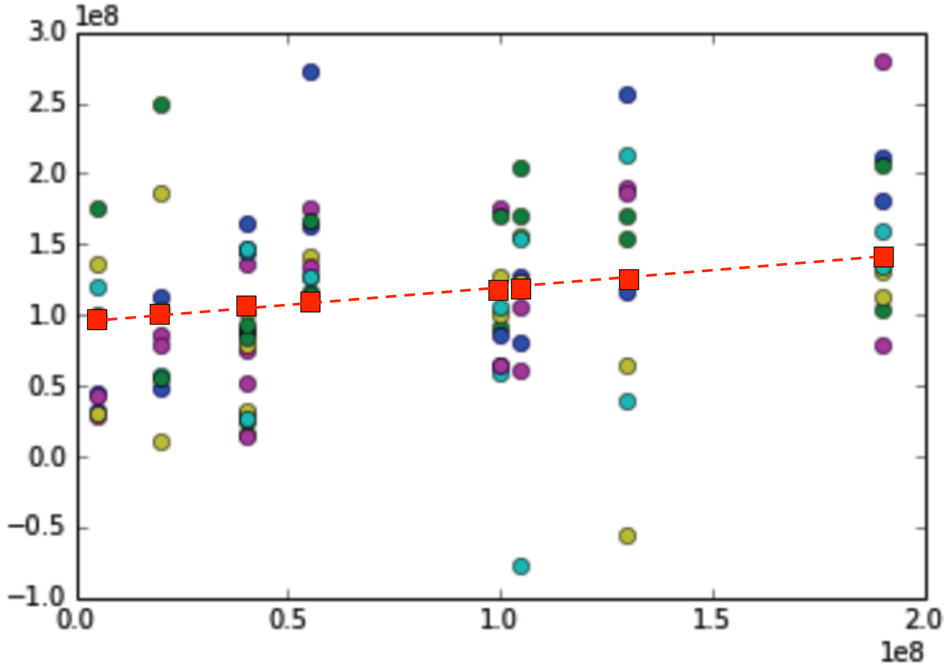
```
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```



Possible Values in alternative universes

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

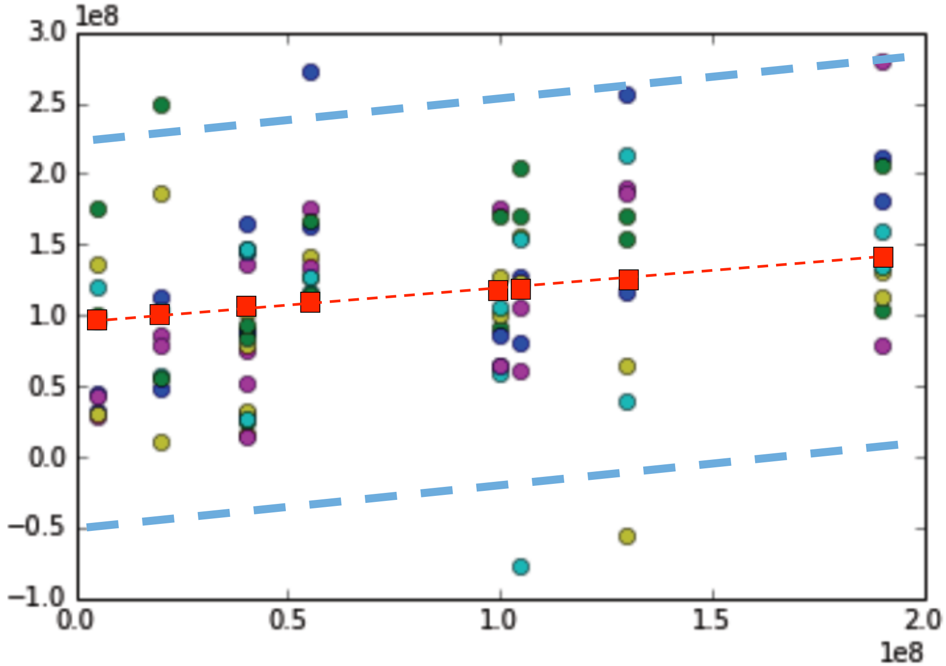
```
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```



Expected value is β0+β1x (without ε)

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

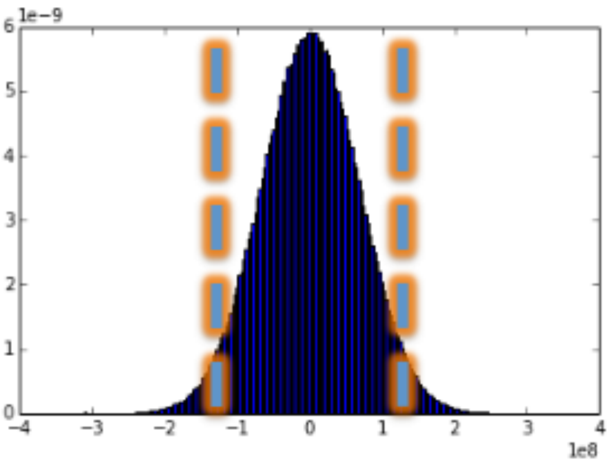
```
def underlying_gross_model(budget):
   return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```

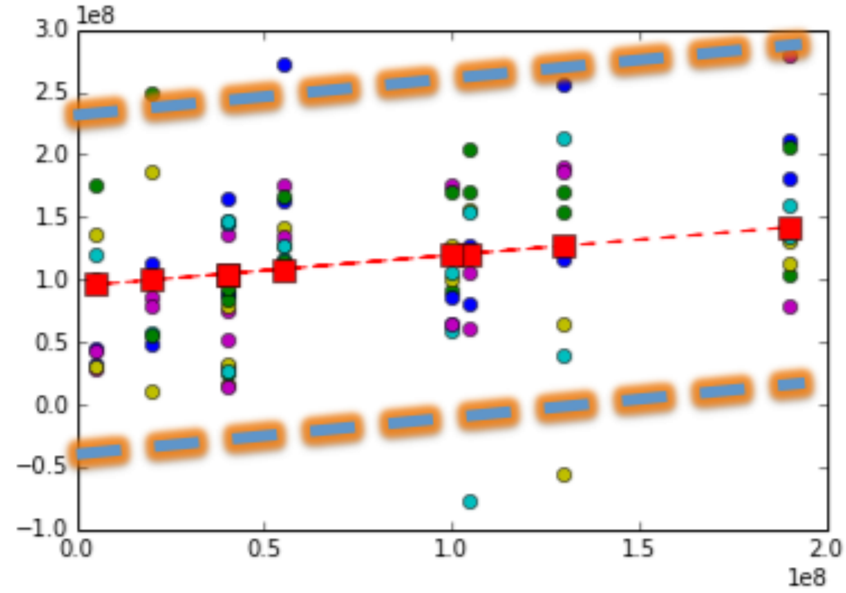


95% prediction interval

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

```
def underlying_gross_model(budget):
  return 94.68e6 + 0.248*budget +random.gauss(0,67762000)
```

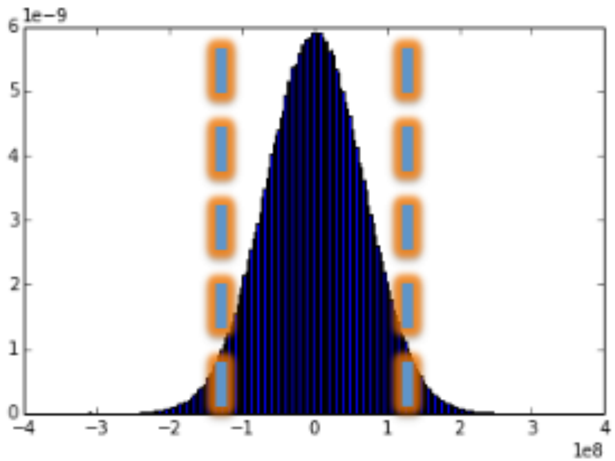


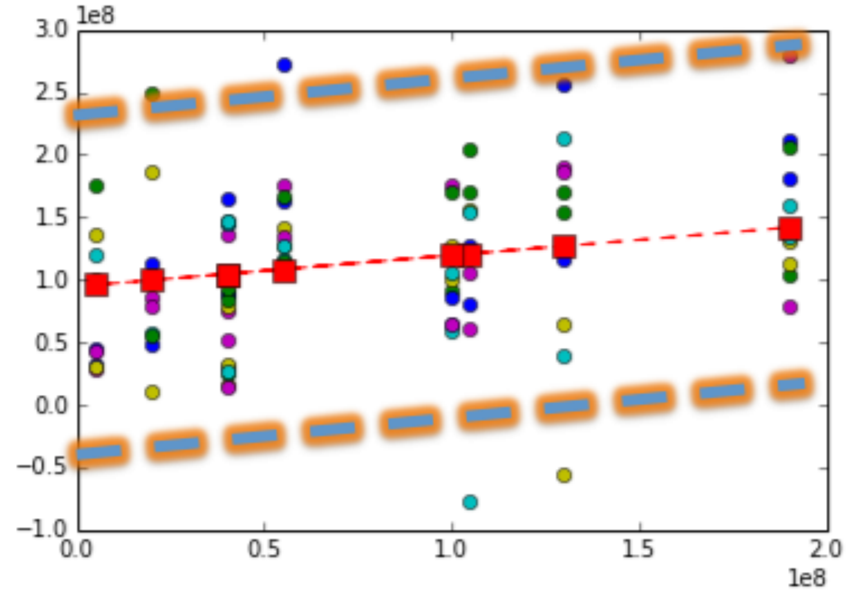


95% prediction interval

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

```
def underlying_gross_model(budget):
  return random.gauss(94.68e6 + 0.248*budget, 67762000)
```





95% prediction interval

Multiple Linear Regression

DATA SCIENCE BOOTCAMP

$$y_\beta(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \varepsilon$$

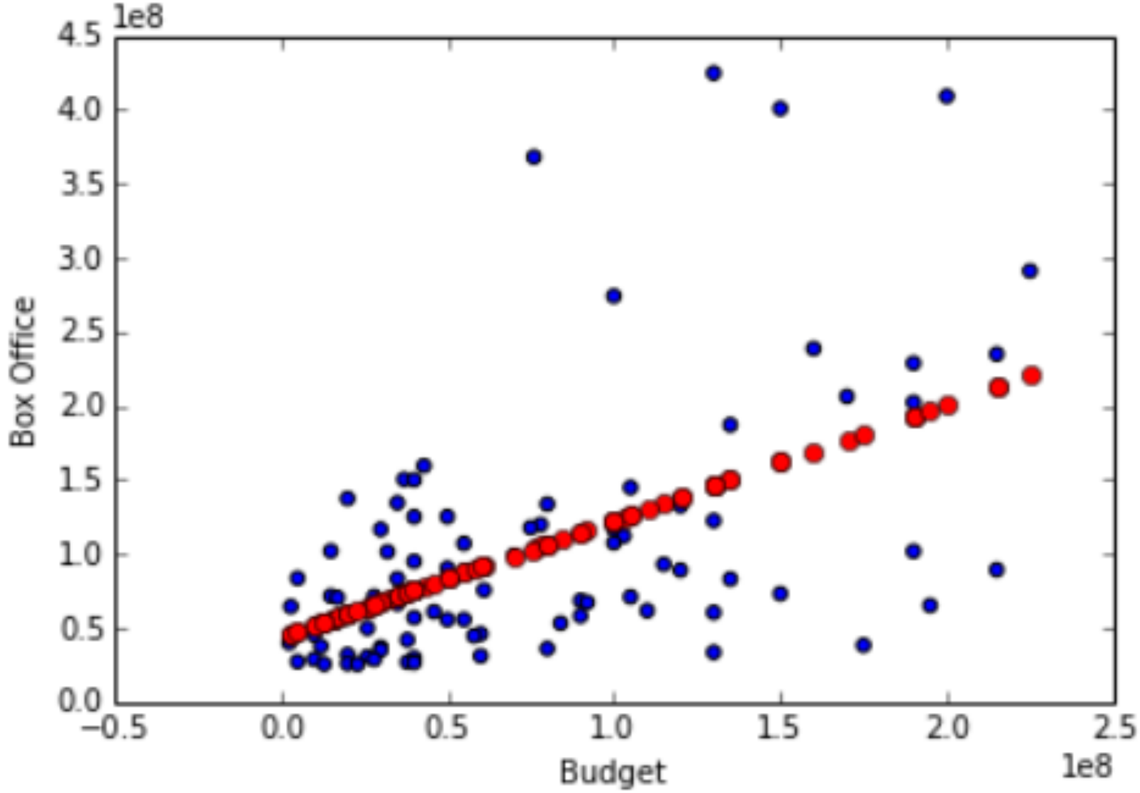
$$y_\beta(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \varepsilon$$

$$\min J(\beta_0, \beta_1, \beta_2, \beta_3, \beta_4)$$

to find the best fitting model

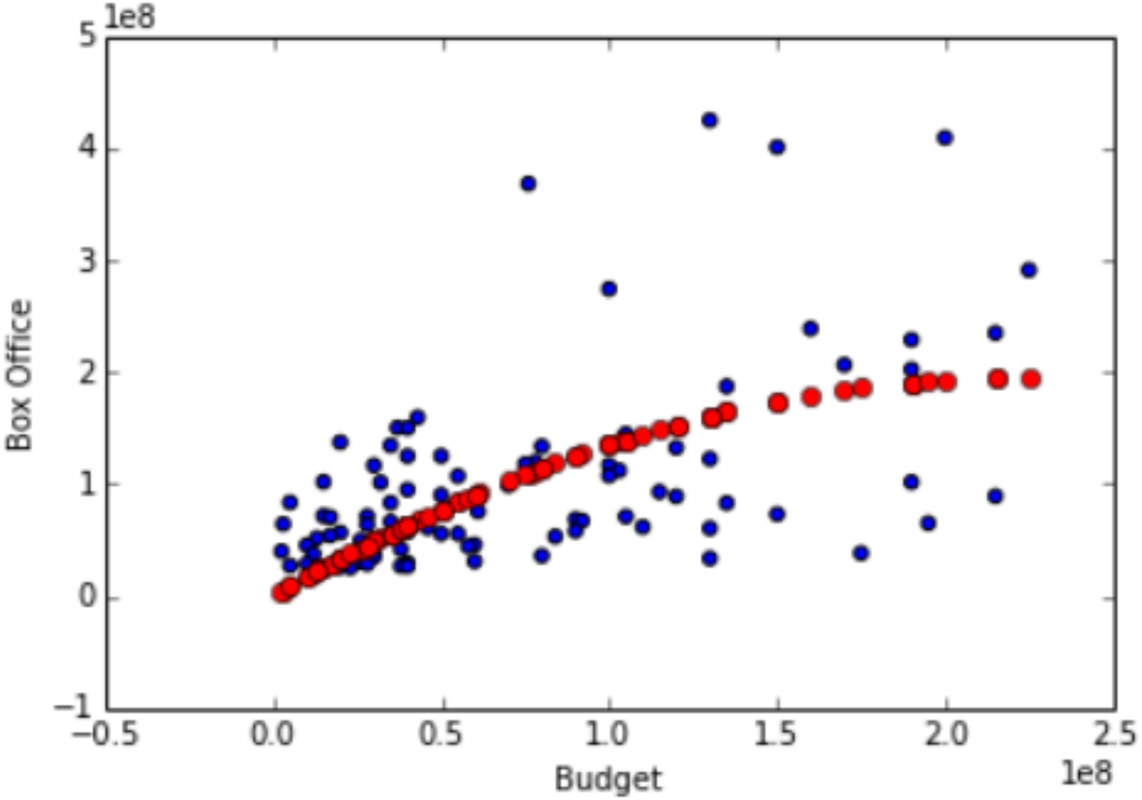
Polynomial regression

$$y_{\beta}(x) = \beta_0 + \beta_1 x + \varepsilon$$



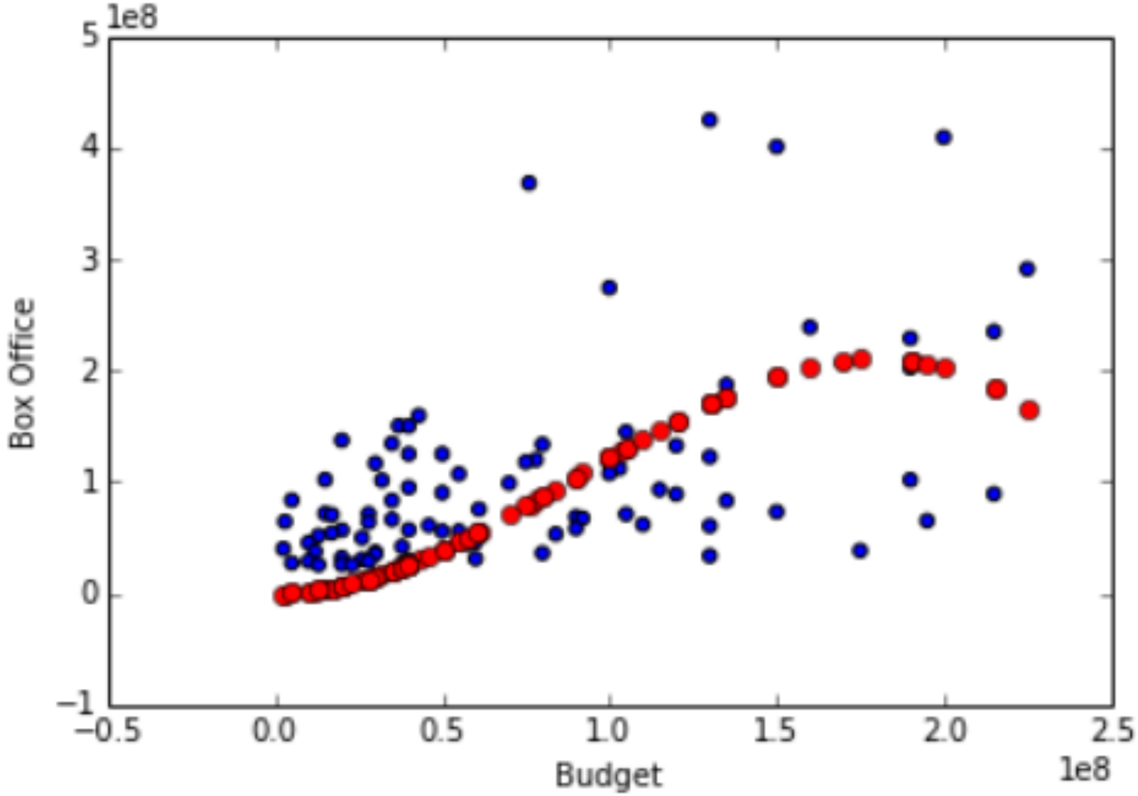
Polynomial regression

$$y_\beta(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon$$



Polynomial regression

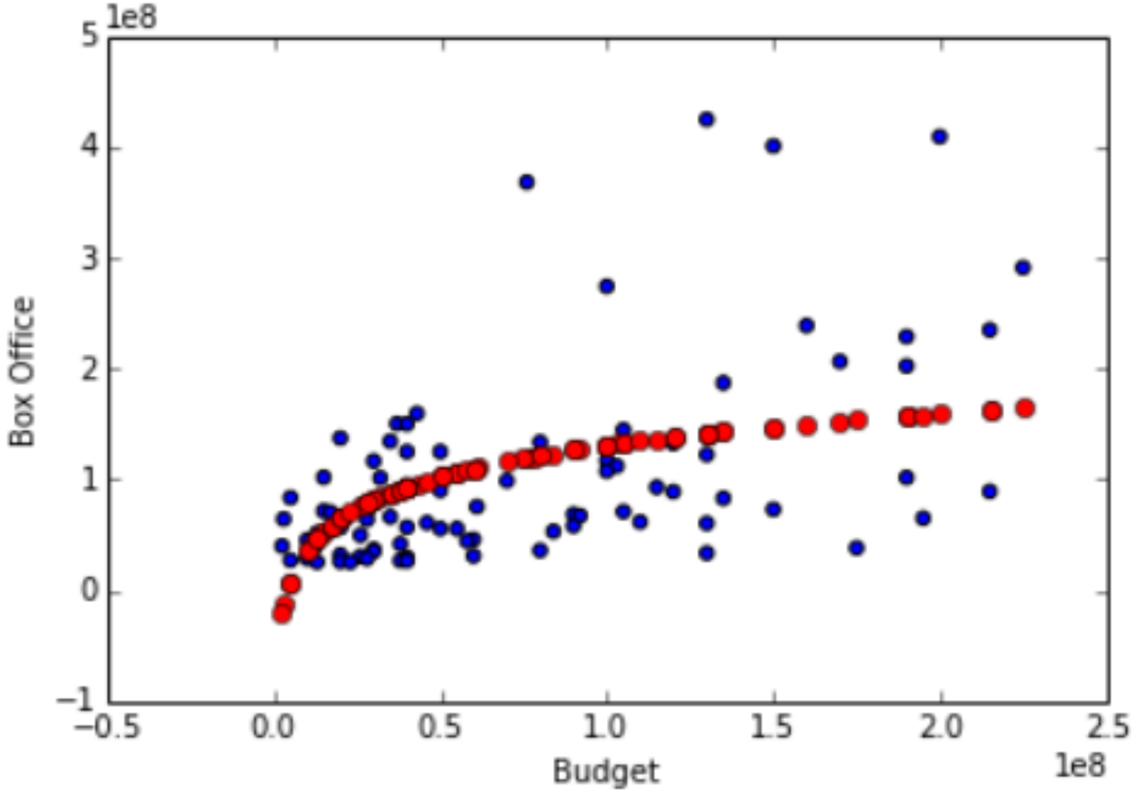
$$y_\beta(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \varepsilon$$



Other functional forms

log

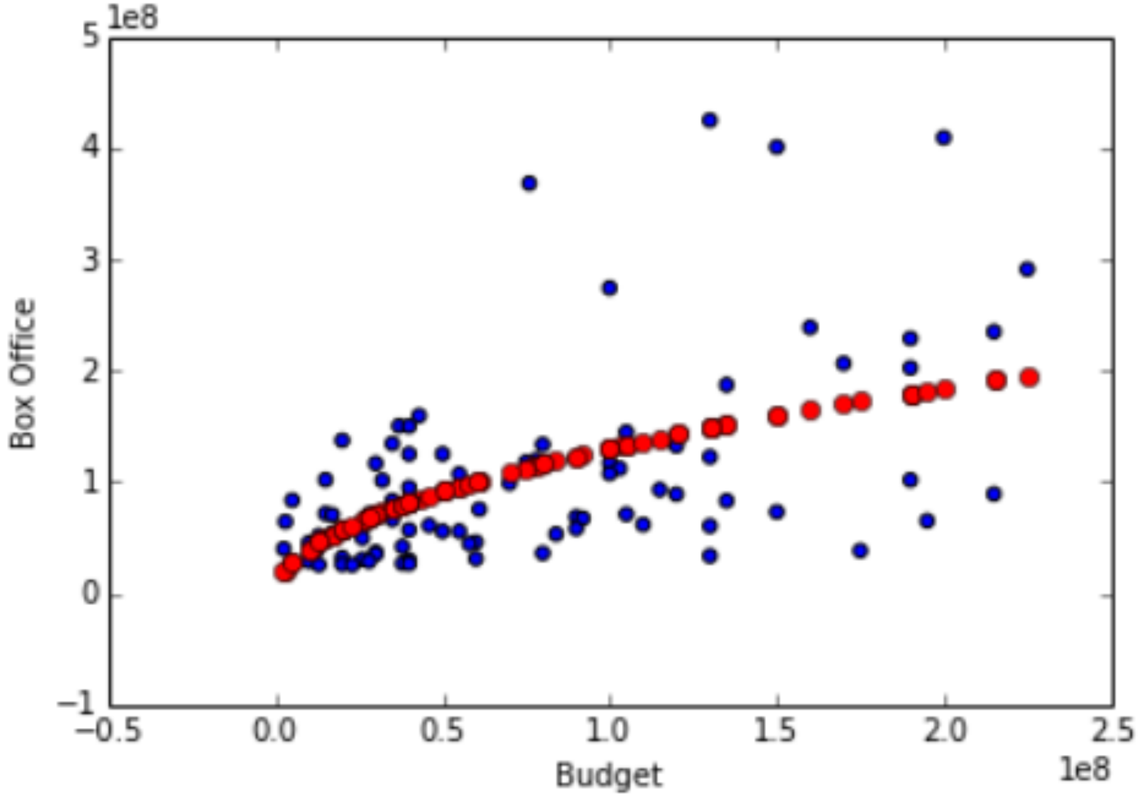
$$y_\beta(x) = \beta_0 + \beta_1 \log(x) + \varepsilon$$



Other functional forms

square root

$$y_\beta(x) = \beta_0 + \beta_1\sqrt{x} + \varepsilon$$



Possible to combine variables

$$y_\beta(x) = \beta_0 + \beta_1 \exp(x_1) + \beta_2 x_2 + \beta_3 x_2^2 + \beta_4 \log(x_3) + \varepsilon$$

Possible to combine variables

$$y_\beta(x) = \beta_0 + \beta_1 \exp(x_1) + \beta_2 x_2 + \beta_3 x_2^2 + \beta_4 \log(x_3) + \varepsilon$$

Interactions

(example: existence of both genres has an each extra effect, different than the sum of each)

$$y_\beta(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \varepsilon$$

$$y_\beta(x) = \beta_0 + \beta_1 \exp(x_1) + \beta_2 x_2 + \beta_3 x_2^2 + \beta_4 \log(x_3) + \varepsilon$$

Linear Regression is not "linear" because we're fitting "a line."

We also fit many other forms.

It's "linear" because the features are combined in a linear fashion ($\Sigma\beta_i f(x_i)$).

Linear

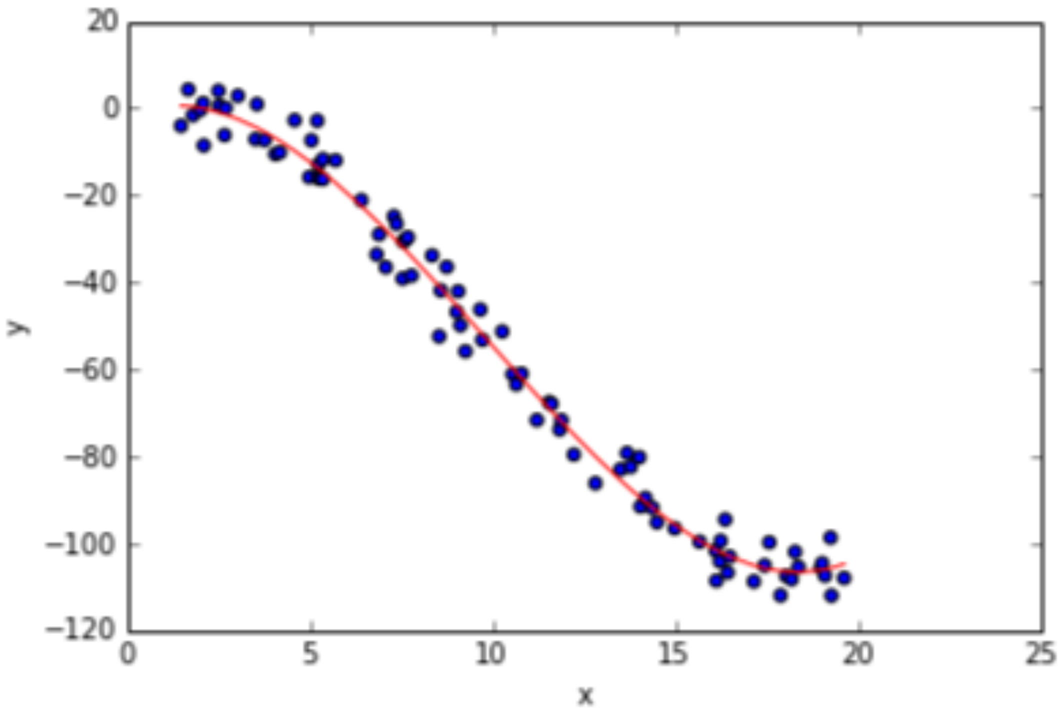
$$y_\beta(x) = \beta_0 + \beta_1 \exp(x_1) + \beta_2 x_2^{-1} + \varepsilon$$

Nonlinear

$$y_\beta(x) = \beta_0 + \beta_1 e^{\beta_2 x_1} + \frac{\beta_3 x_2}{(1 + \beta_4 x_2)} + \varepsilon$$

How to choose functional forms to try?

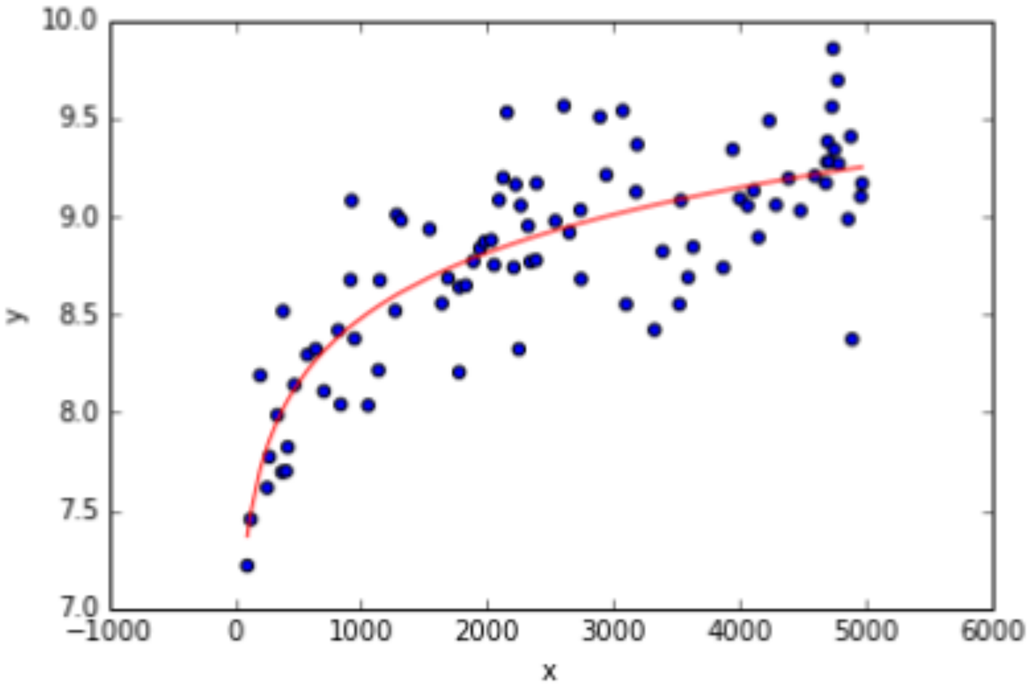
Check one on one relationship of variable with outcome



$$y_\beta(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \varepsilon$$

How to choose functional forms to try?

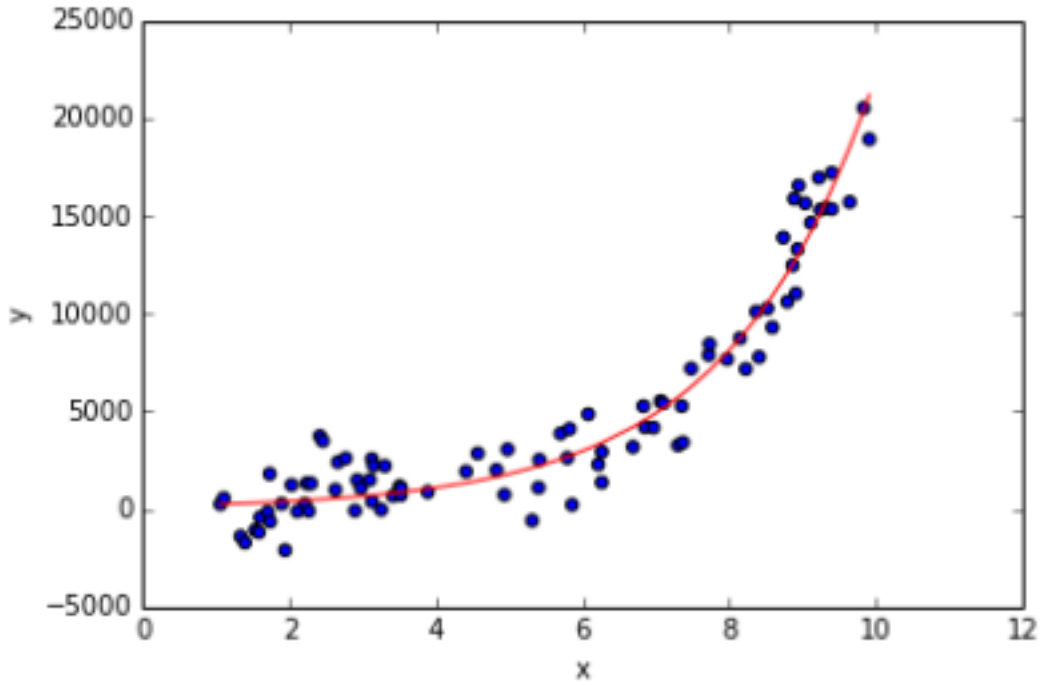
Check one on one relationship of variable with outcome



$$y_\beta(x) = \beta_0 + \beta_1 \log(x) + \varepsilon$$

How to choose functional forms to try?

Check one on one relationship of variable with outcome



$$\log(y_\beta(x)) = \beta_0 + \beta_1 x + \varepsilon$$