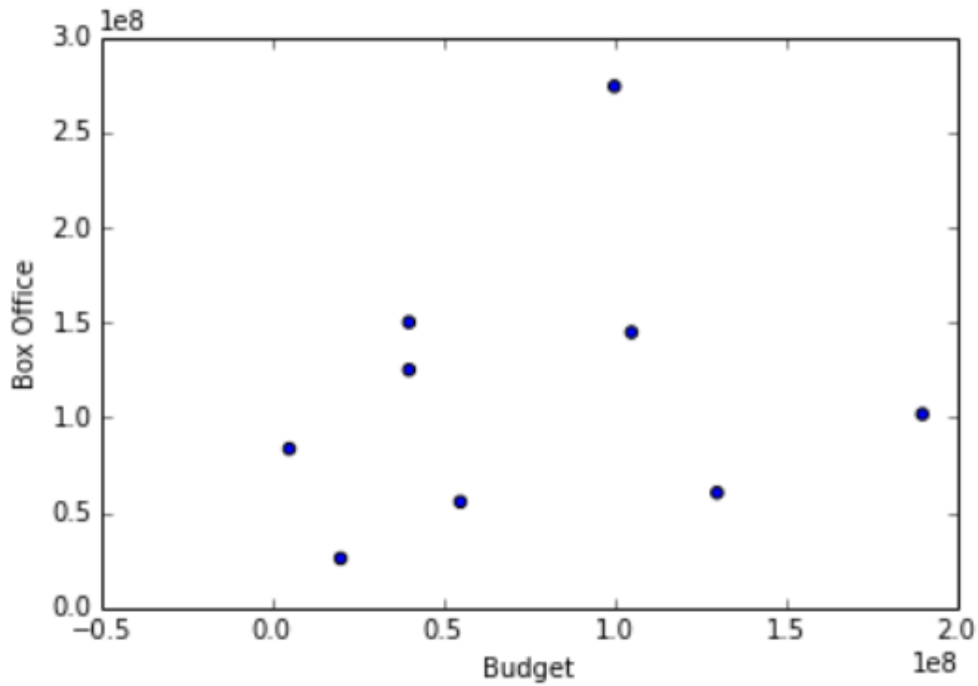
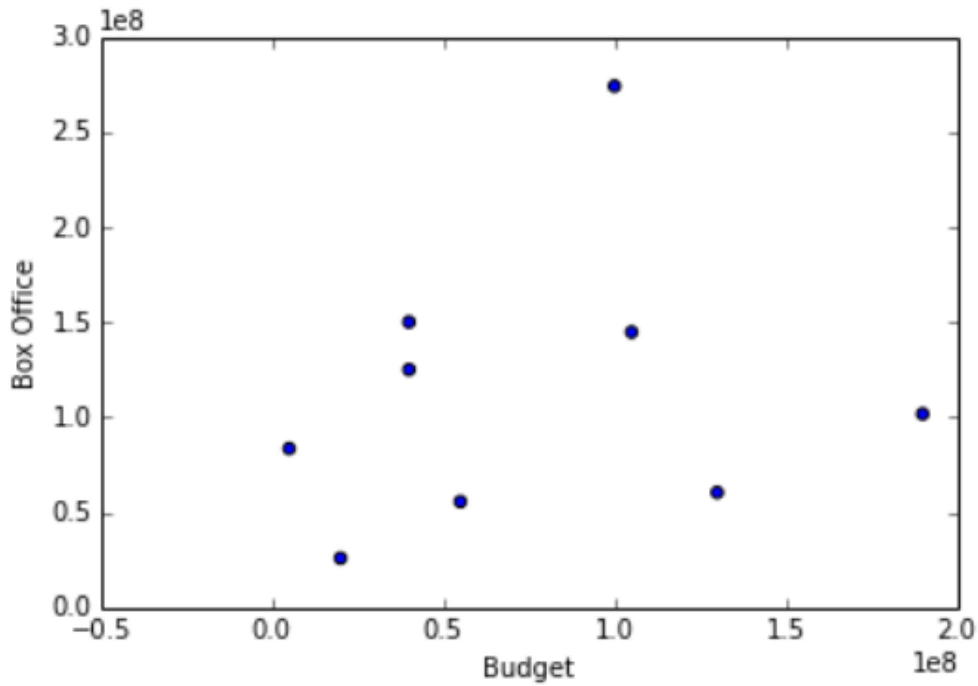
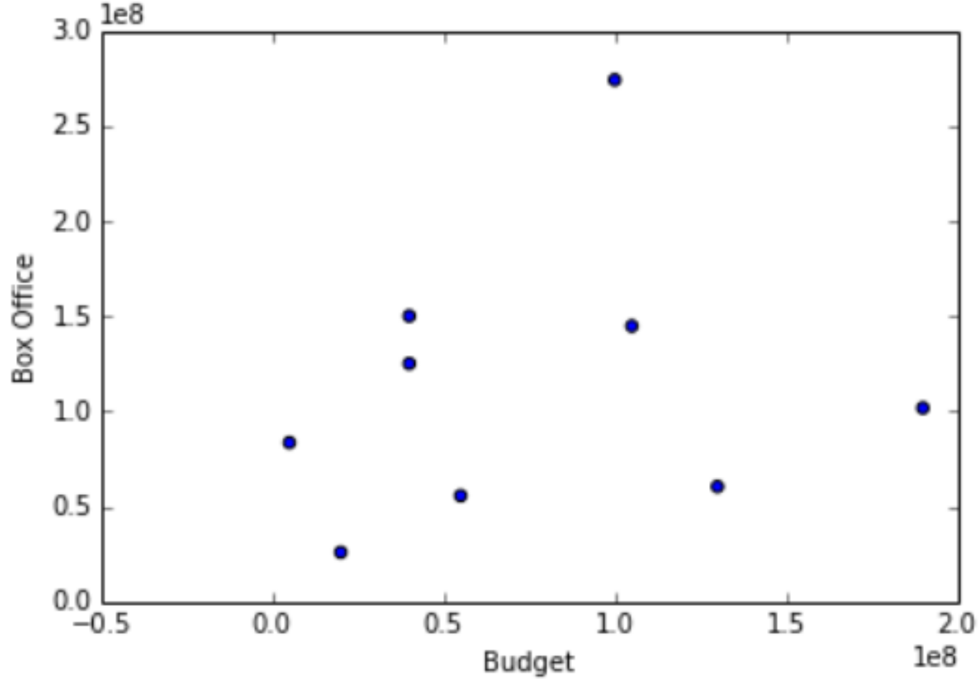
# Linear Regression


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




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

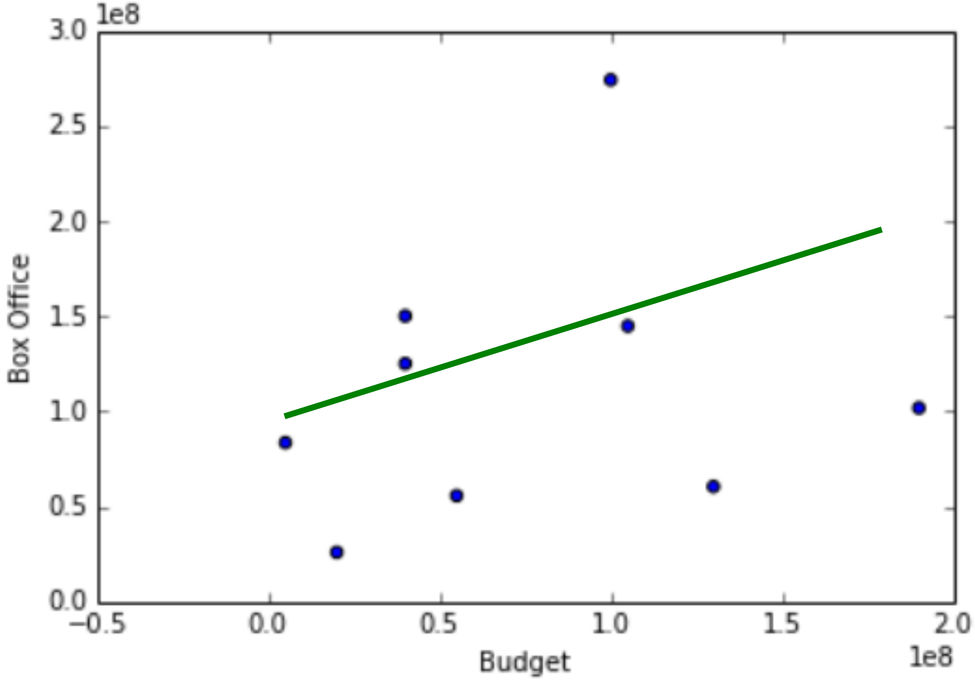



coef 0 coef 1

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

Gross
of
movie

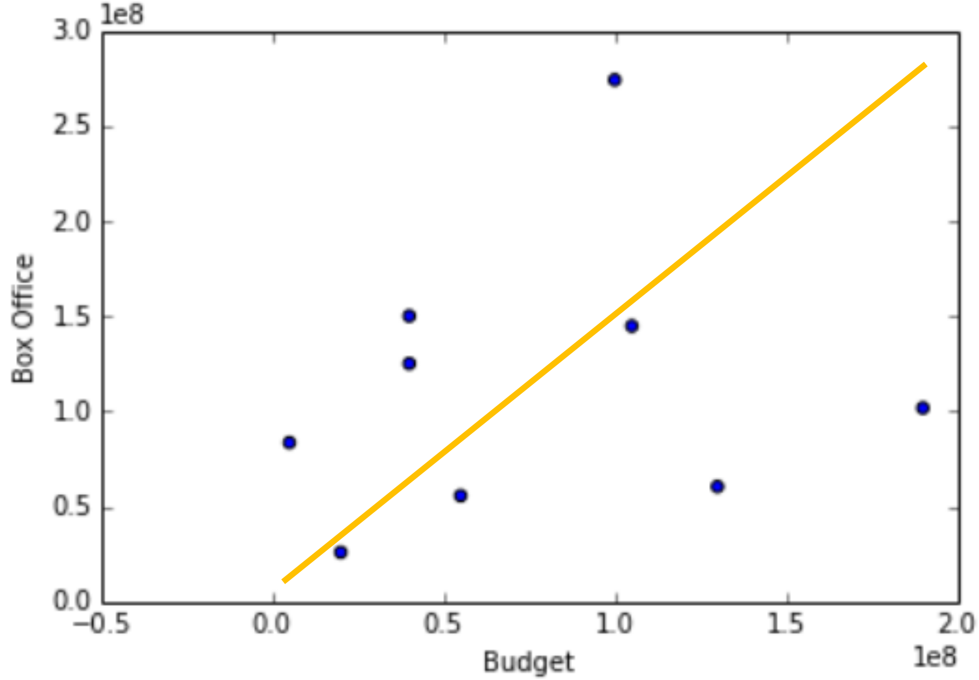
Budget
of
movie




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

$\beta_0 = 80 million$

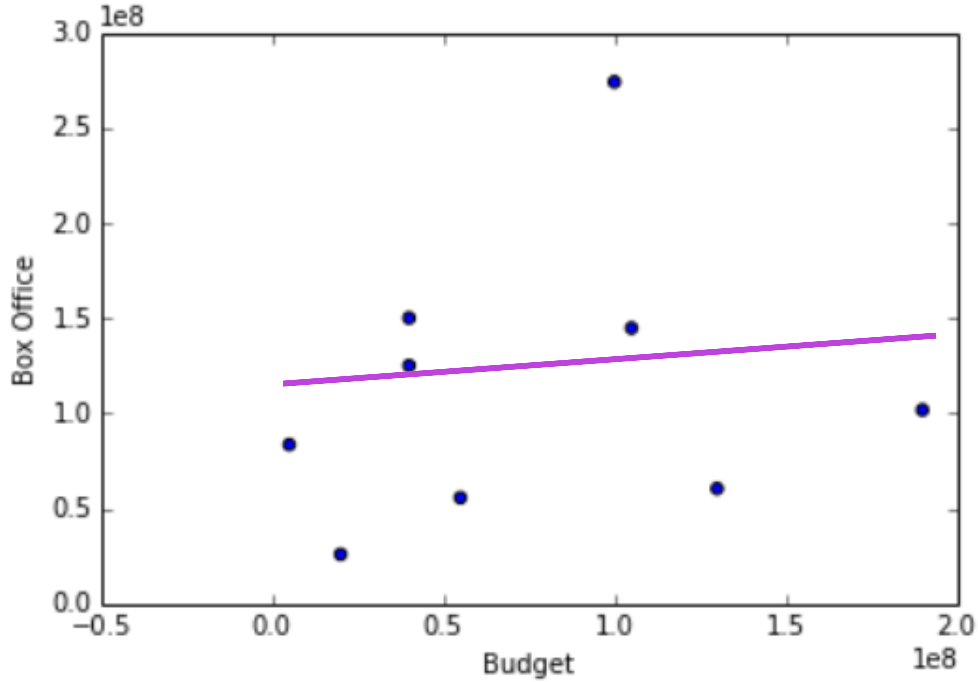
$\beta_1 = 0.5$




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

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

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

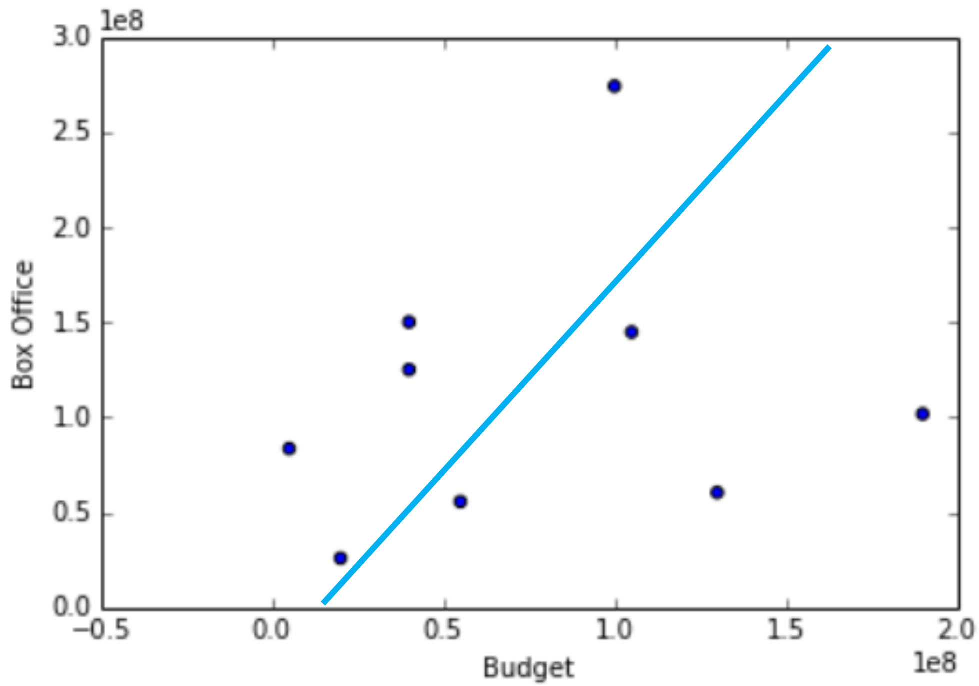



$$y_\beta(x) = \beta_0 + \beta_1 x$$

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

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

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



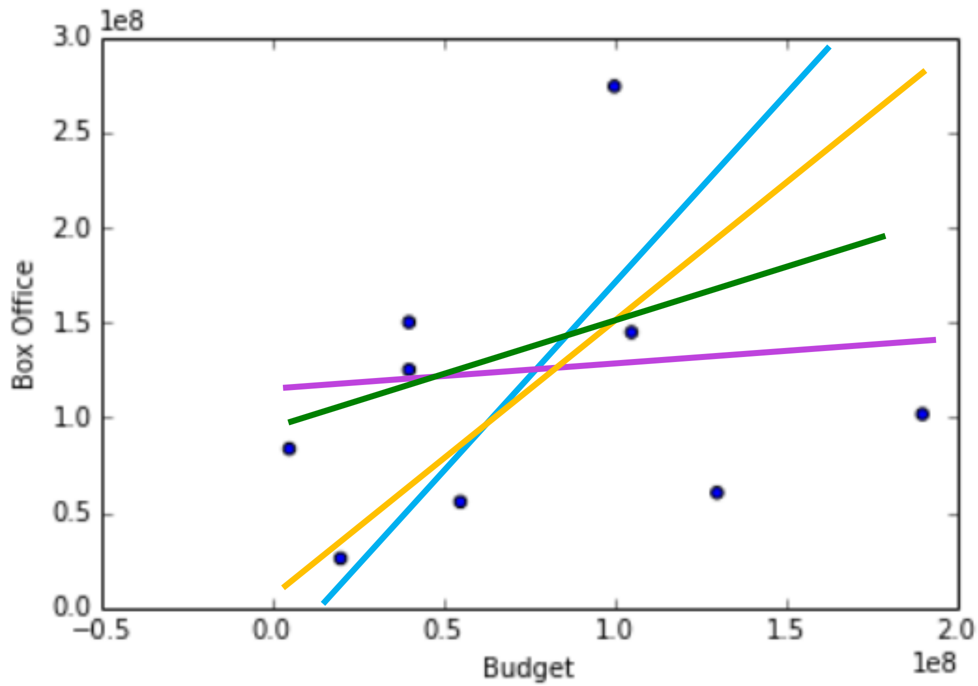

$$y_{\beta}(x) = \beta_0 + \beta_1 x$$

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

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

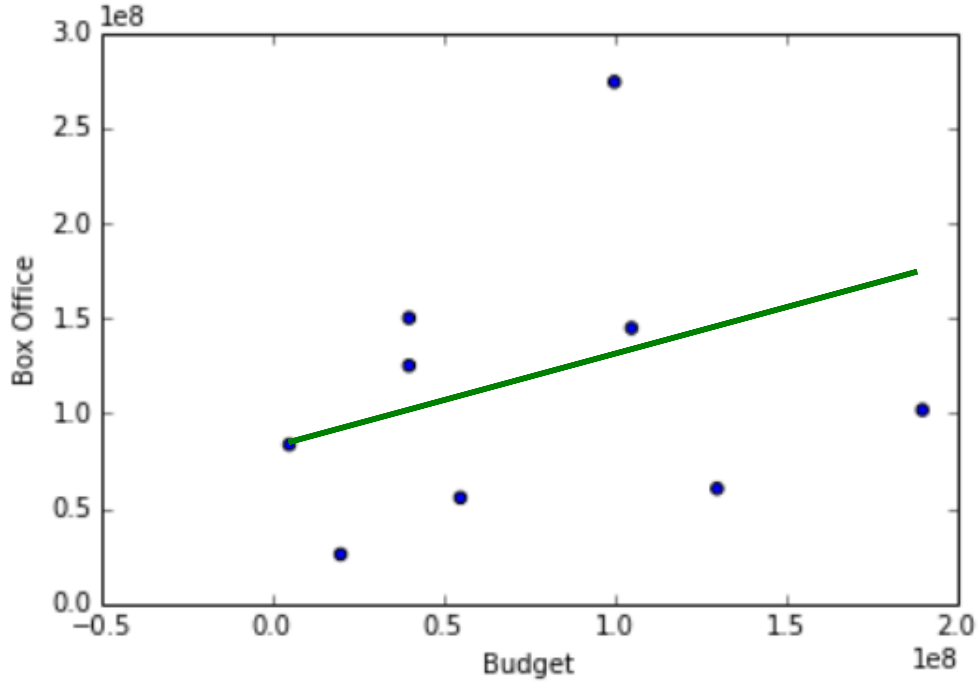
$\beta_0 = 120 million$
$\beta_1 = 0.1$

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



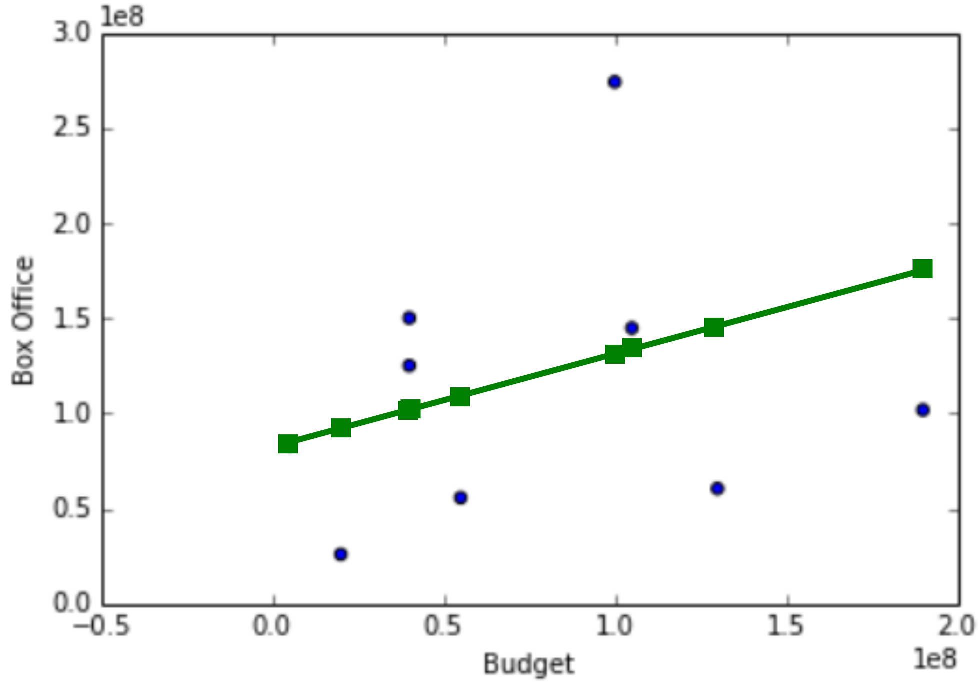

$$y_\beta(x) = \beta_0 + \beta_1 x$$

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



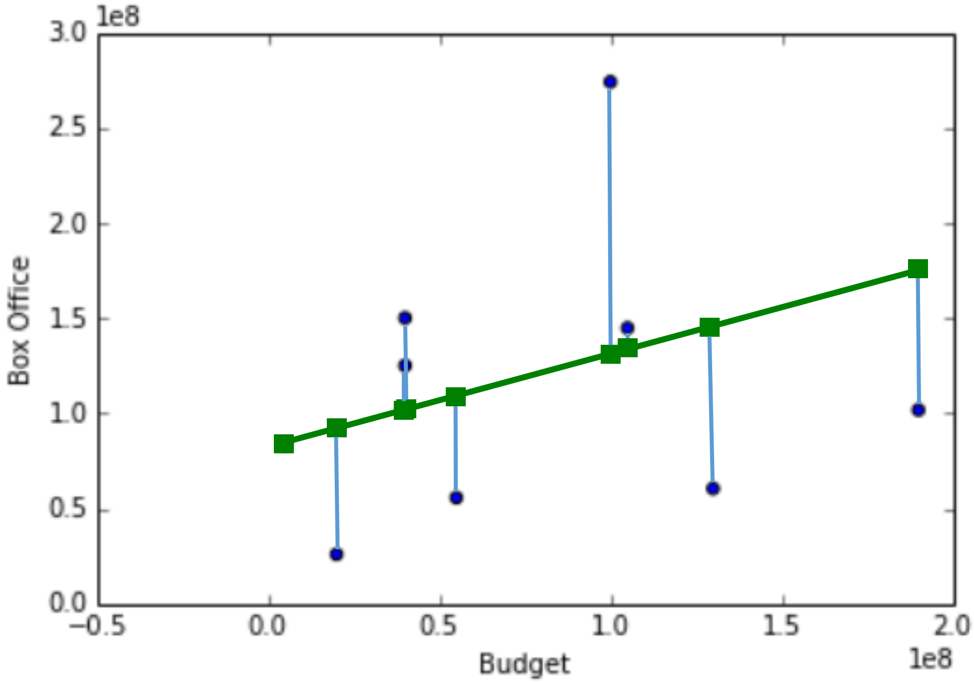

$$y_\beta(x) = \beta_0 + \beta_1 x$$

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




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

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




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

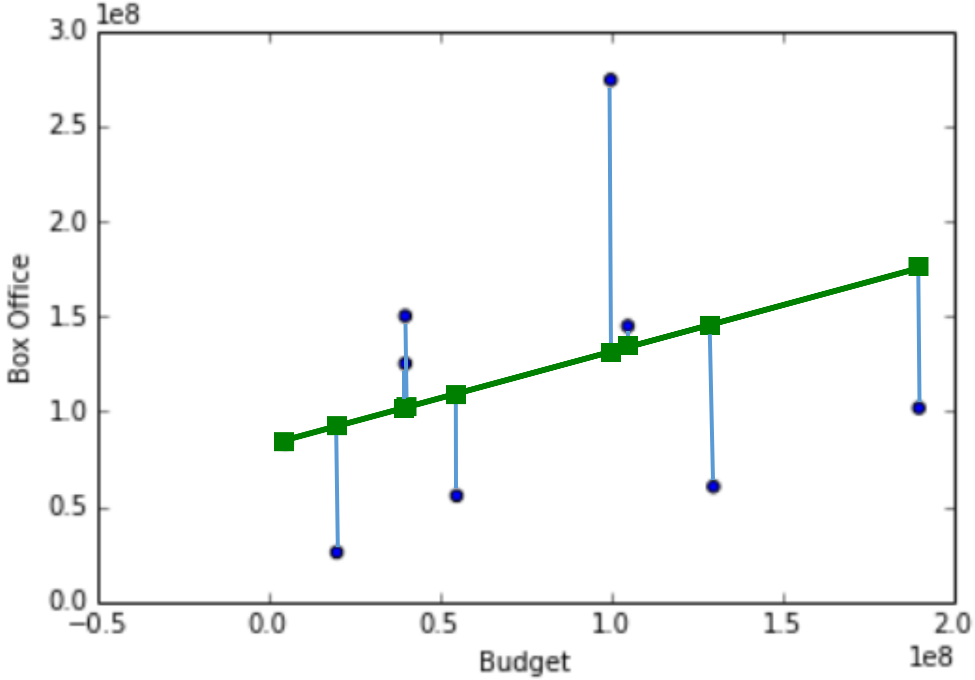
$$y_\beta(x_{obs}^{(1)}) - y_{obs}^{(1)}$$

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

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

Predicted value by model – Observed value

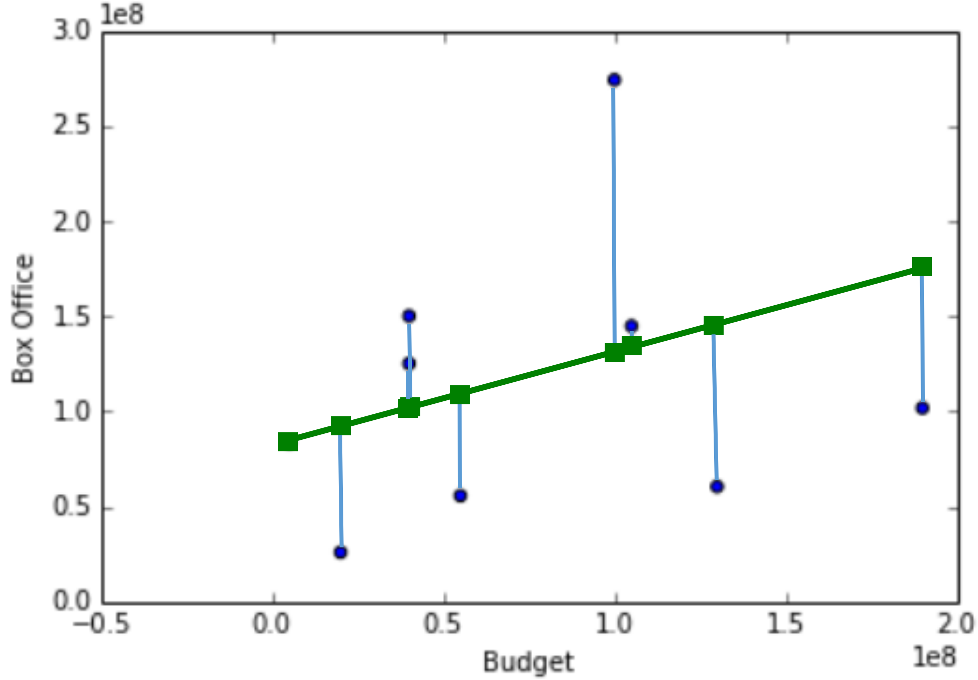
β0 =80M, β1=0.5




Predicted value by model – Observed value

β0 =80M, β1=0.5

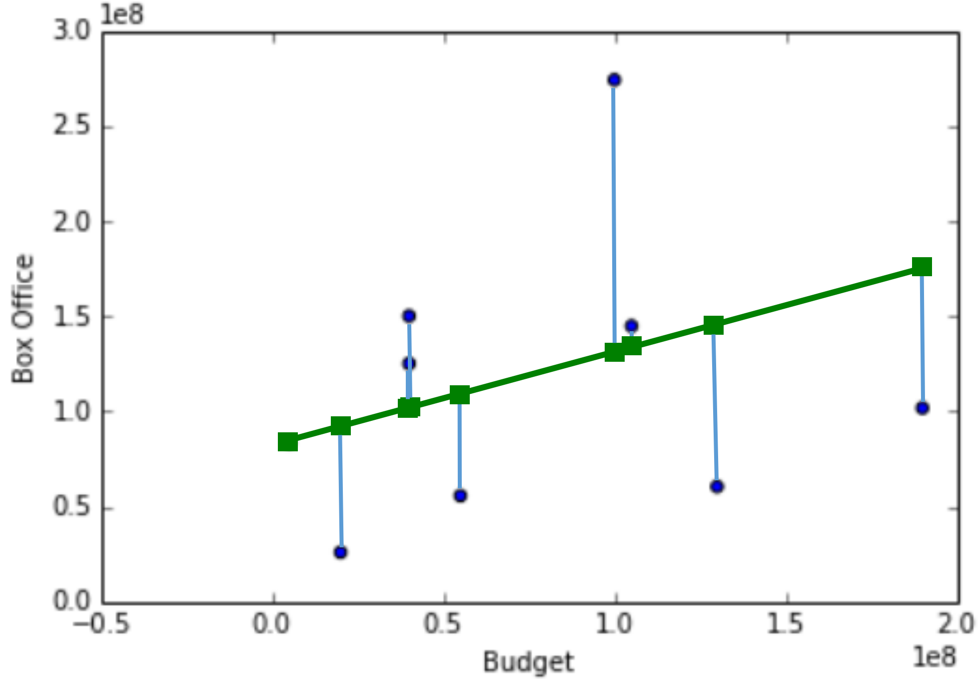
$$y_{\beta}(x_{obs}^{(i)}) - y_{obs}^{(i)}$$




Predicted value by model – Observed value

β0 =80M, β1=0.5

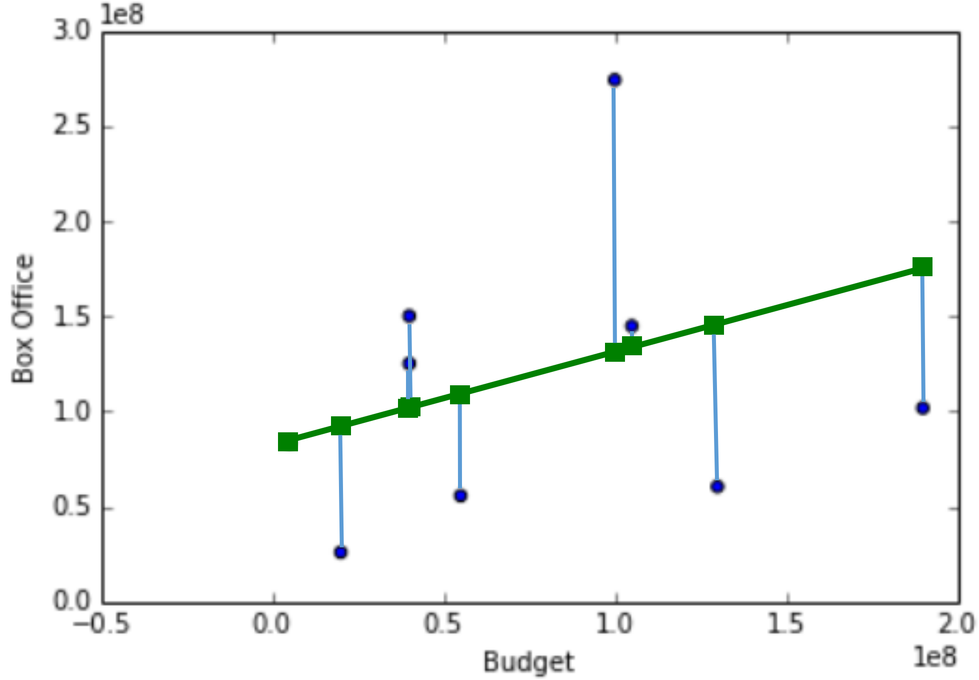
$$(\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)}$$




Predicted value by model – Observed value

β0 =80M, β1=0.5

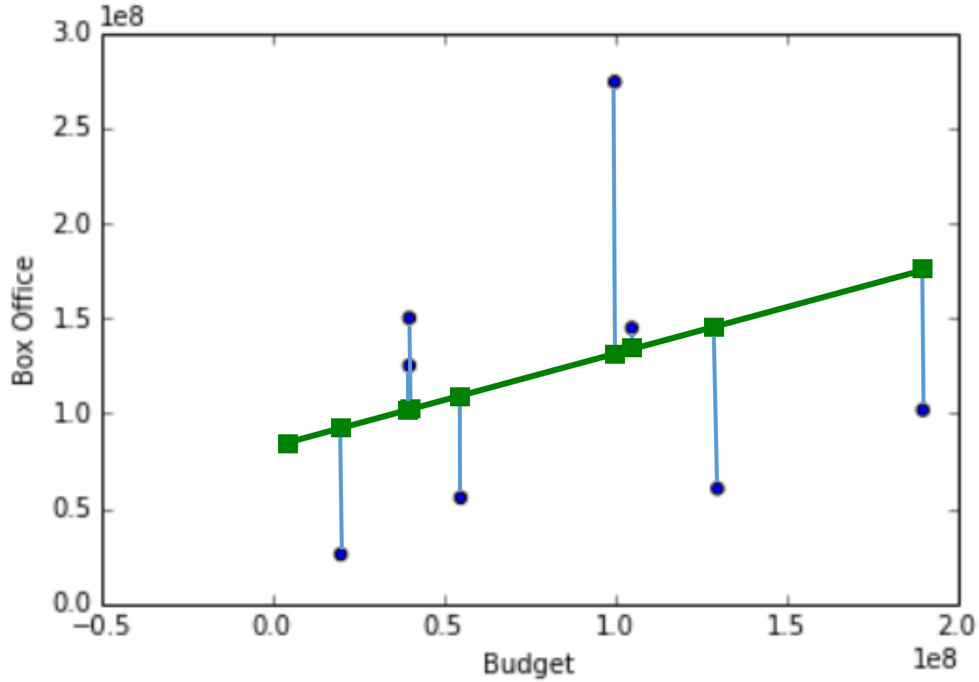
$$\sum_{i=1}^{m}\left((\beta_0+\beta_1 x_{obs}^{(i)})-y_{obs}^{(i)}\right)^2$$




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

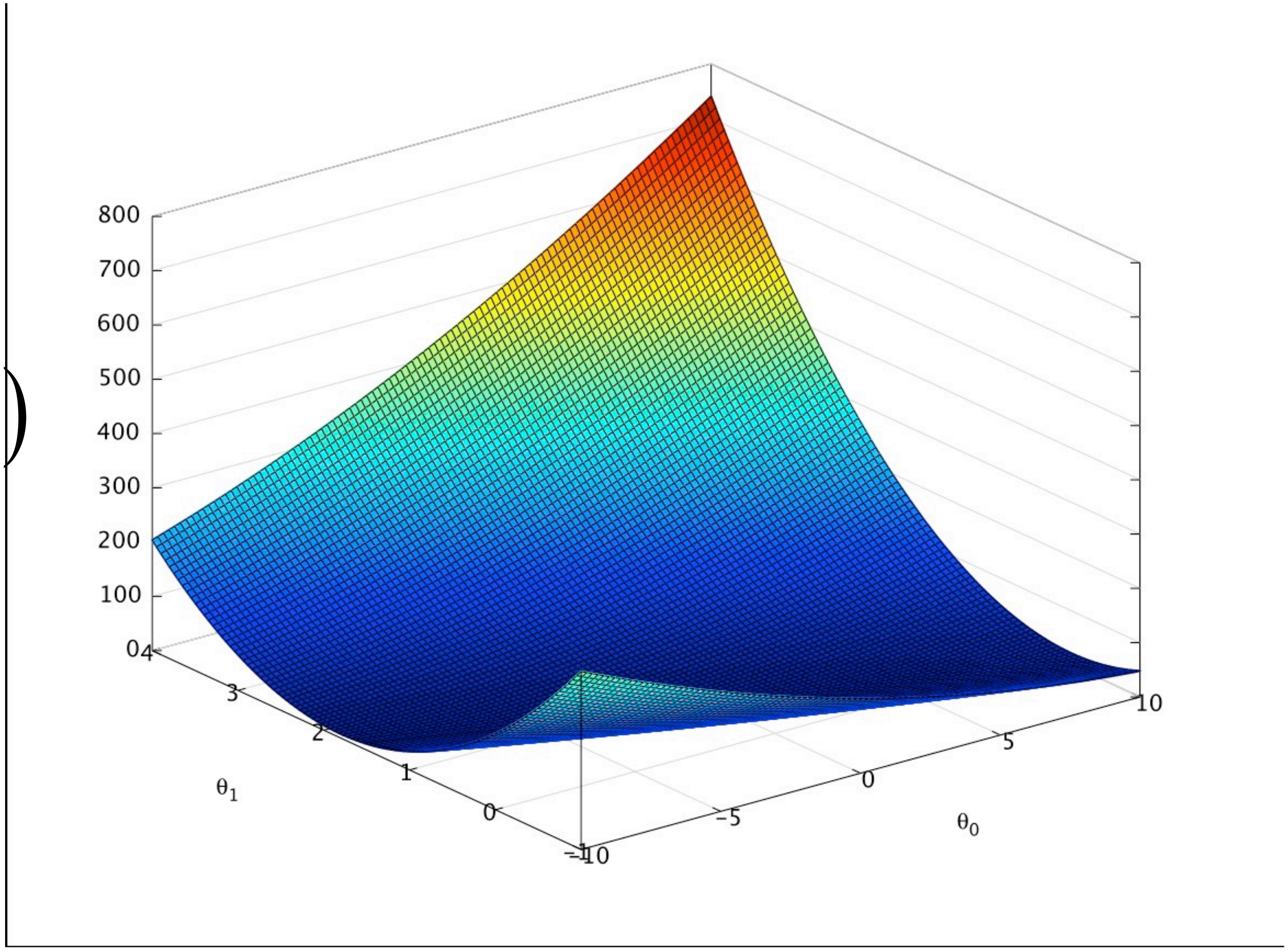
Takes a model (specific parameter values), returns score

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

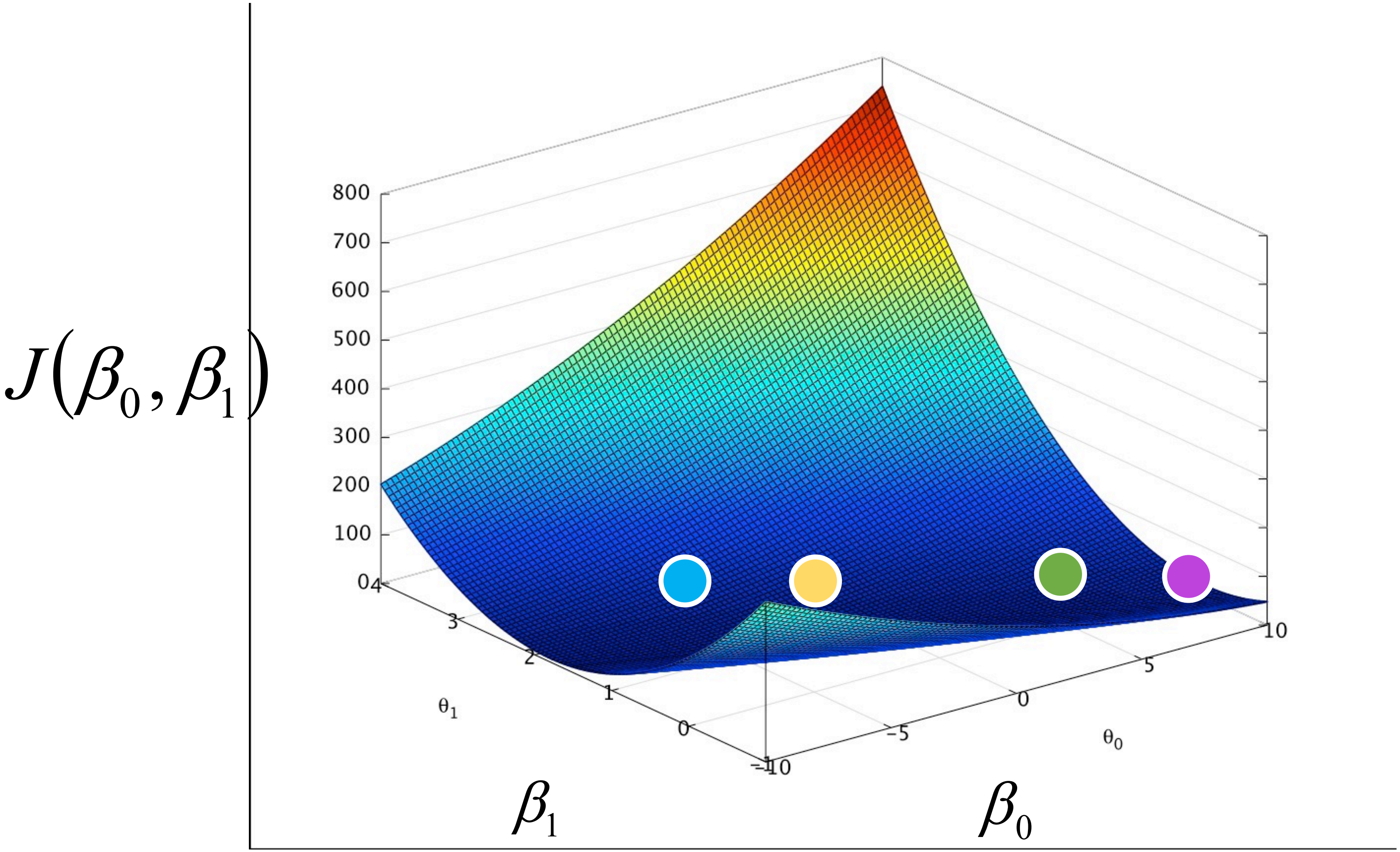
## Cost function

$$J(\beta_0, \beta_1) = \frac{1}{2m} \underbrace{\sum_{i=1}^{m} \left( (\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)} \right)^2}_{\text{Lower for better fits}}$$

$J(\beta_0, \beta_1)$




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

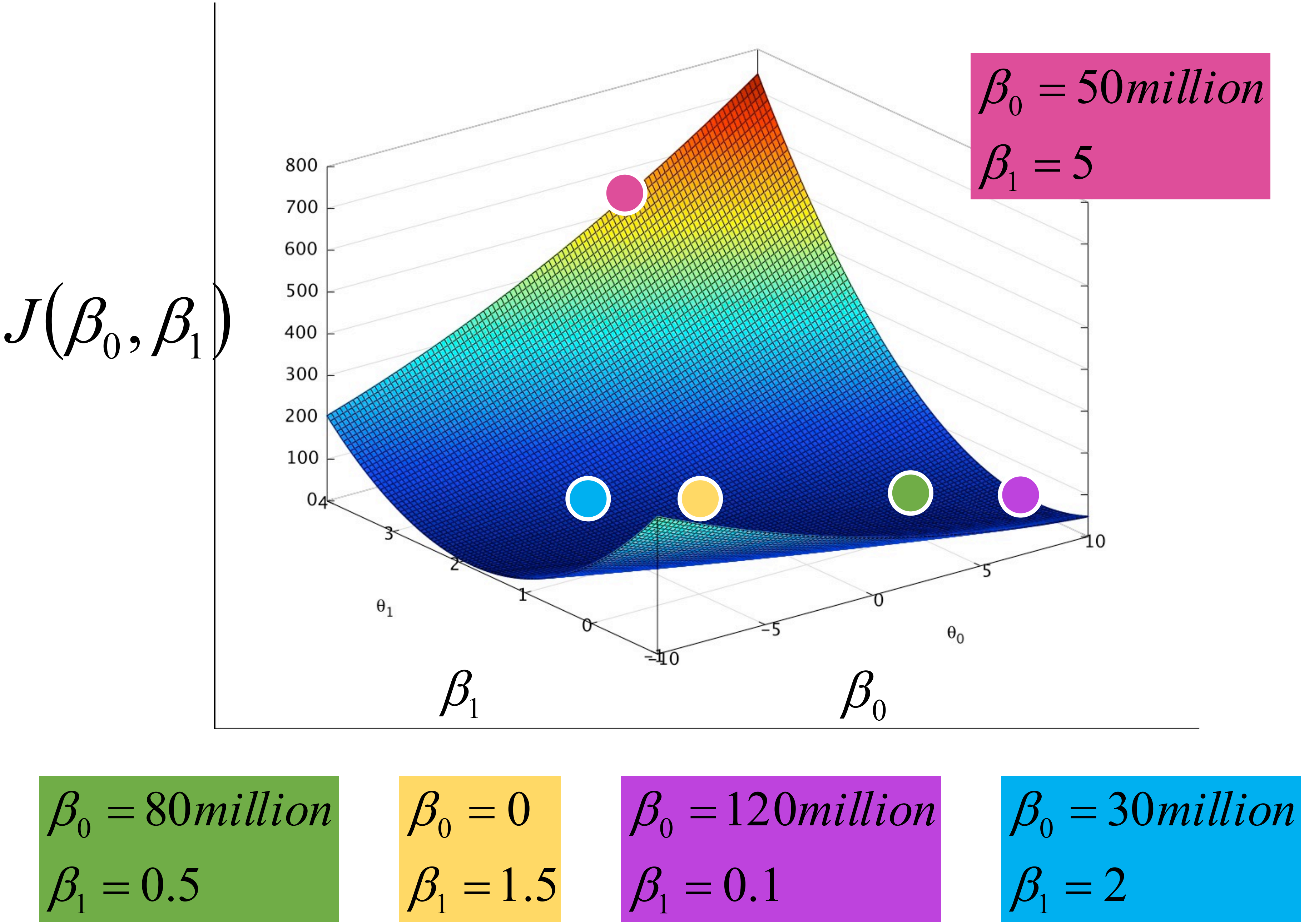



$\beta_0 = 80 million$
$\beta_1 = 0.5$

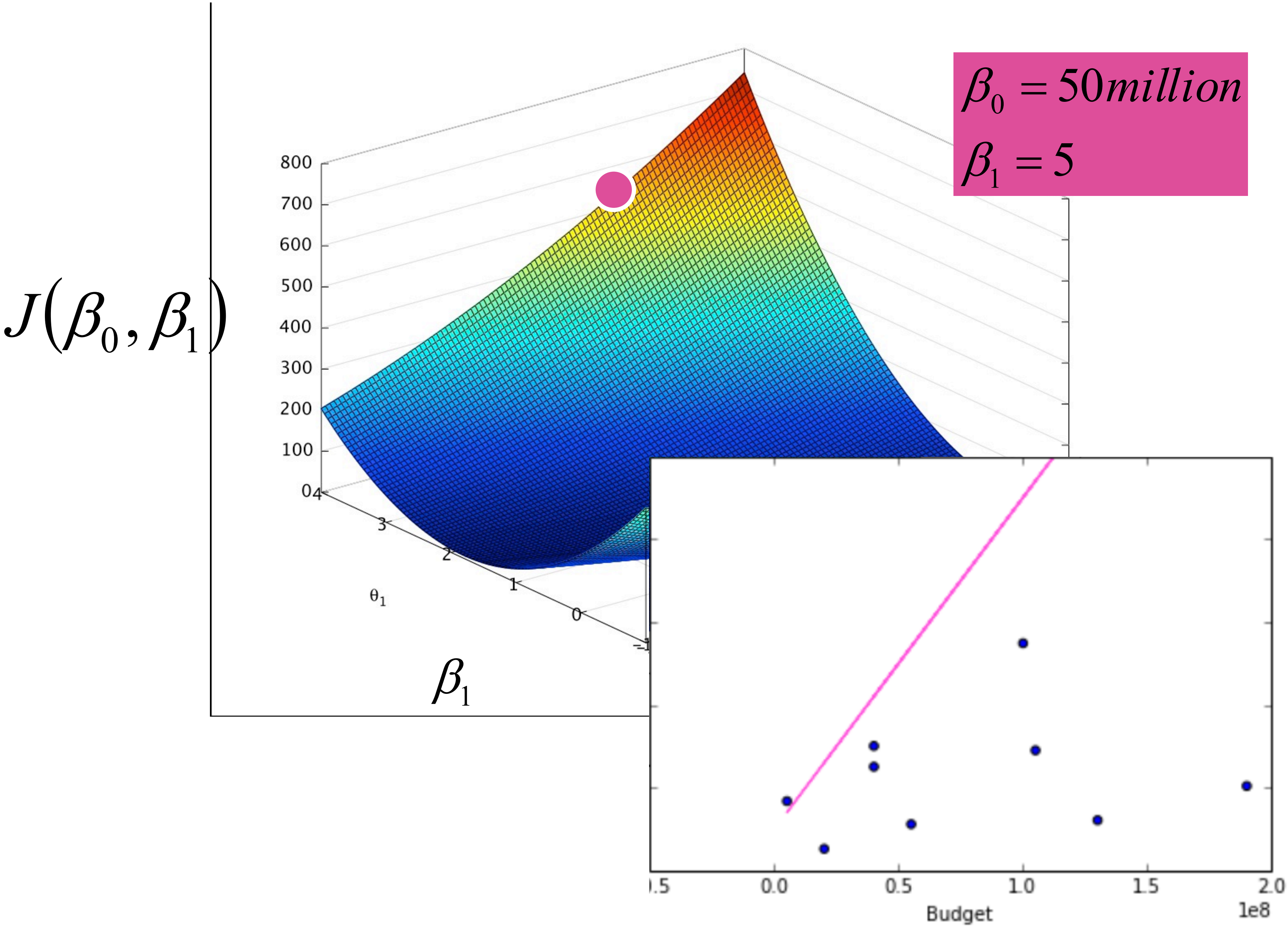
$\beta_0 = 0$
$\beta_1 = 1.5$

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

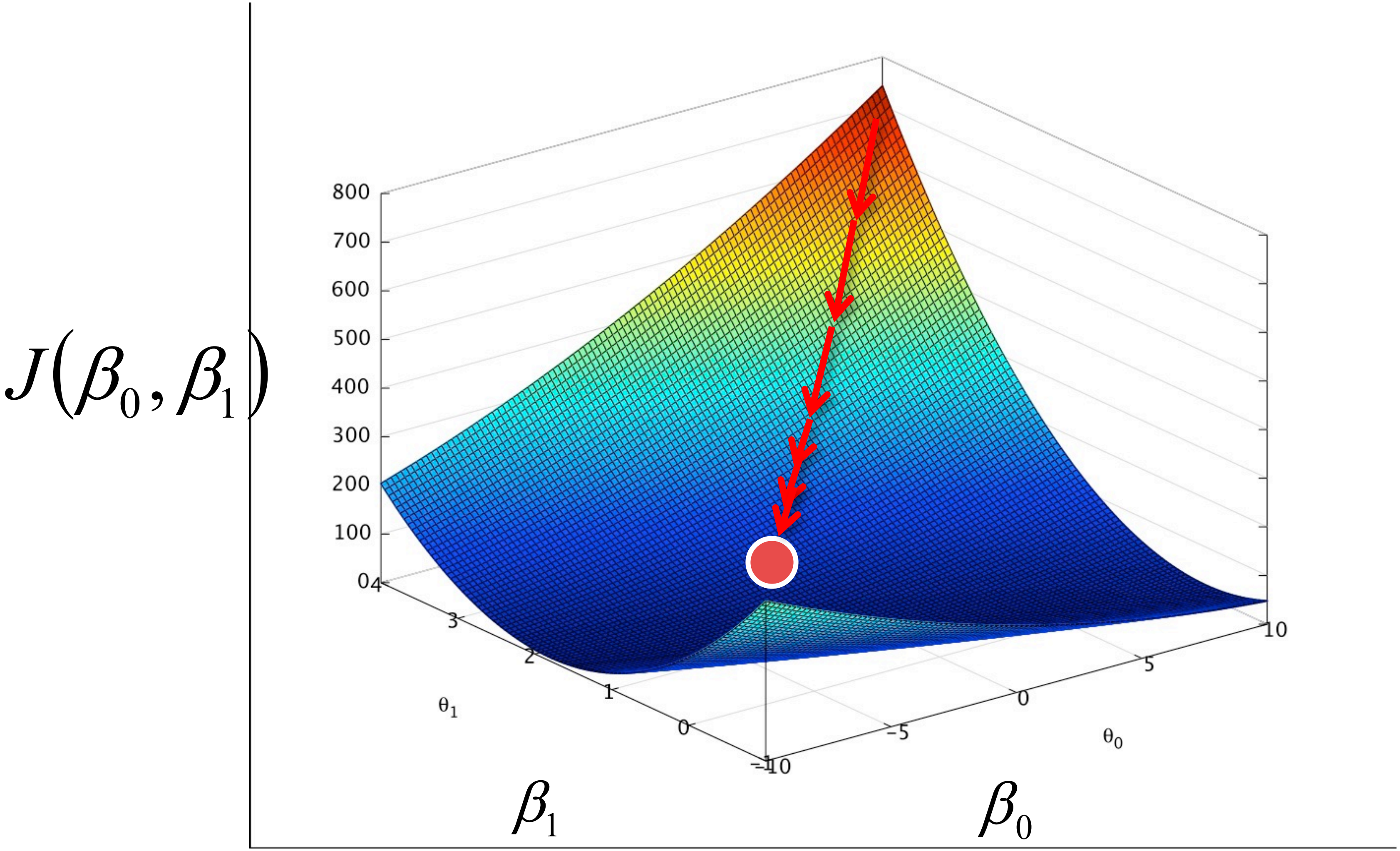
$\beta_0 = 30 million$
$\beta_1 = 2$



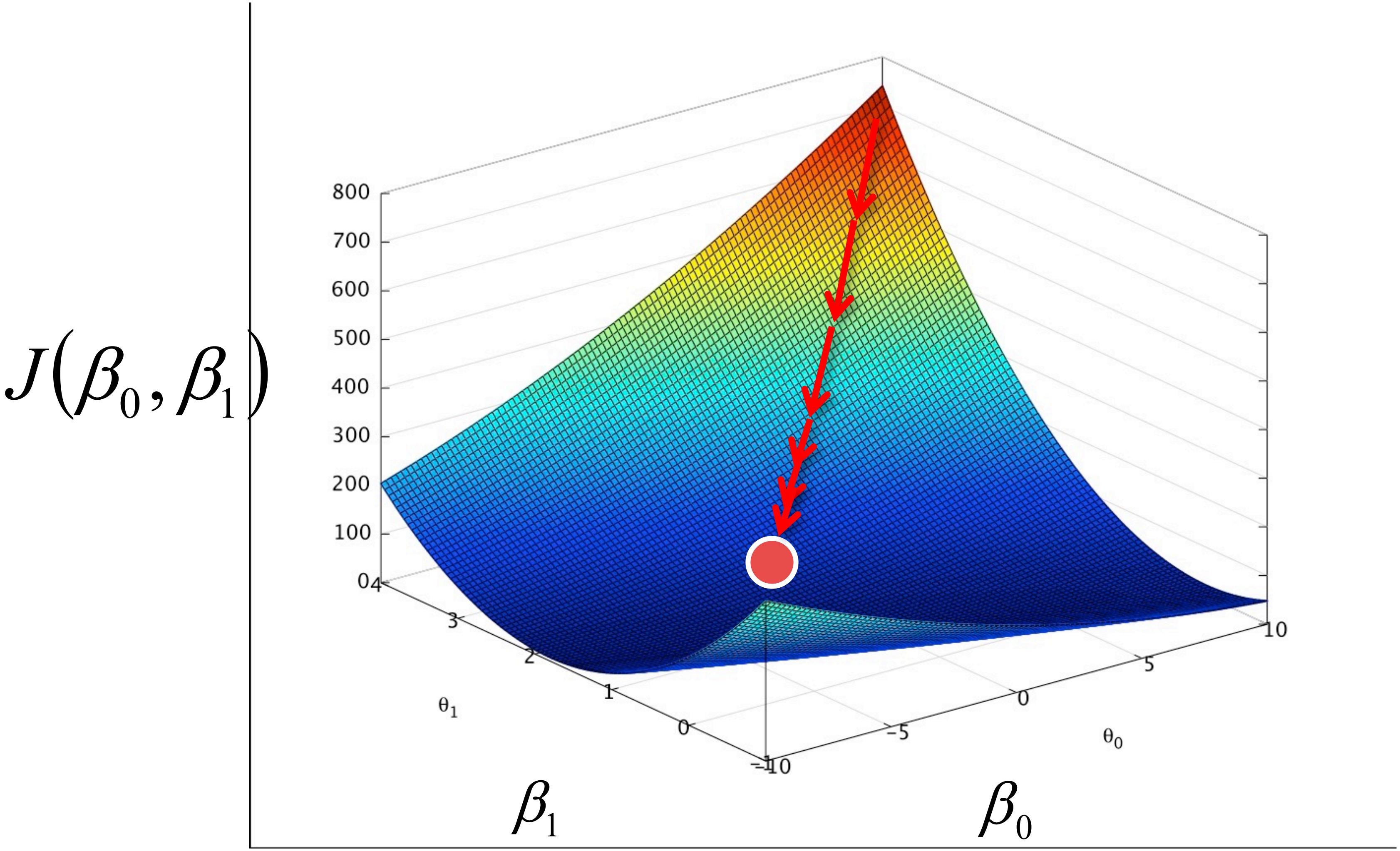
$J(\beta_0, \beta_1)$
800
700
600
500
400
300
200
100
0
$\theta_1$
$\theta_0$
$\beta_1$
$\beta_0$
$\beta_0 = 50 million$
$\beta_1 = 5$
$\beta_0 = 80 million$
$\beta_1 = 0.5$
$\beta_0 = 0$
$\beta_1 = 1.5$
$\beta_0 = 120 million$
$\beta_1 = 0.1$
$\beta_0 = 30 million$
$\beta_1 = 2$



$\beta_0 = 50 million$
$\beta_1 = 5$
$J(\beta_0, \beta_1)$
800
700
600
500
400
300
200
100
0
$\theta_1$
$\beta_1$
0.0
0.5
1.0
1.5
2.0
Budget
1e8

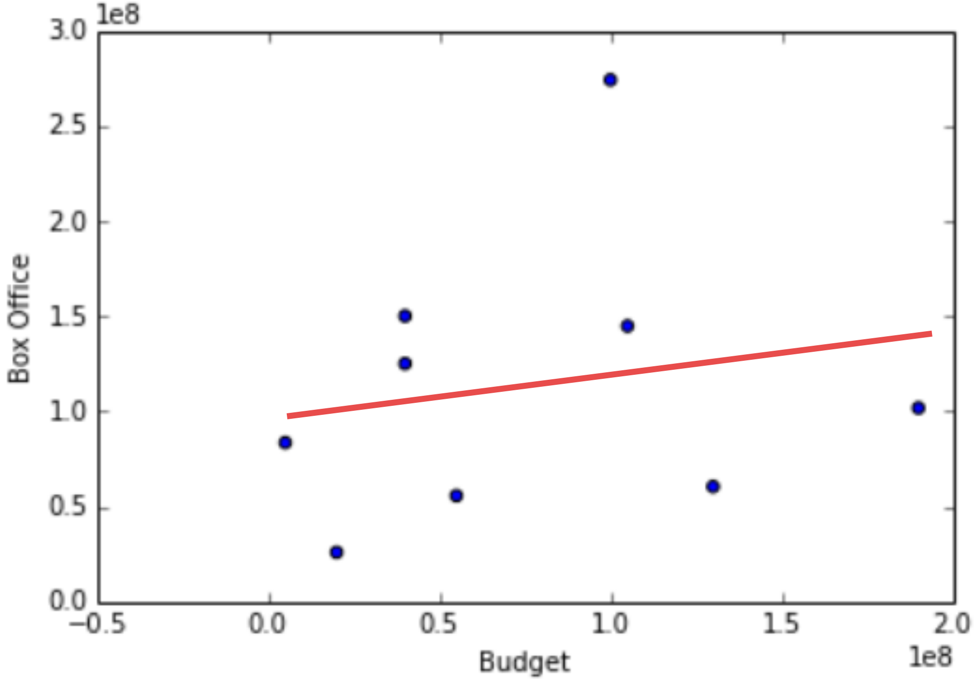



```
import statsmodels.formula.api as sm
linmodel = sm.OLS(Y, X).fit()
```




$\beta_0 = 94.68 million$

$\beta_1 = 0.1$




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

$$\beta_0 = 94.68 million$$

$$\beta_1 = 0.1$$

# Multiple Linear Regression

DATA SCIENCE BOOTCAMP

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

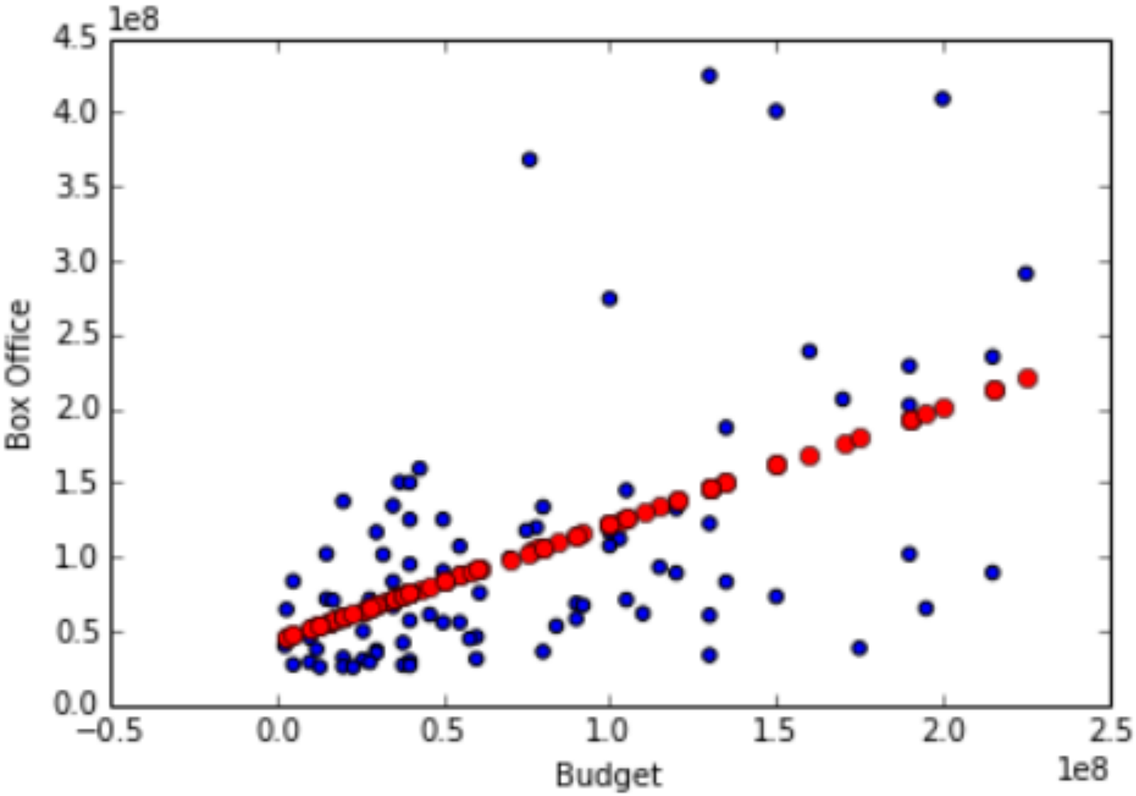
$$y_\beta(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$$

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

to find the best fitting model

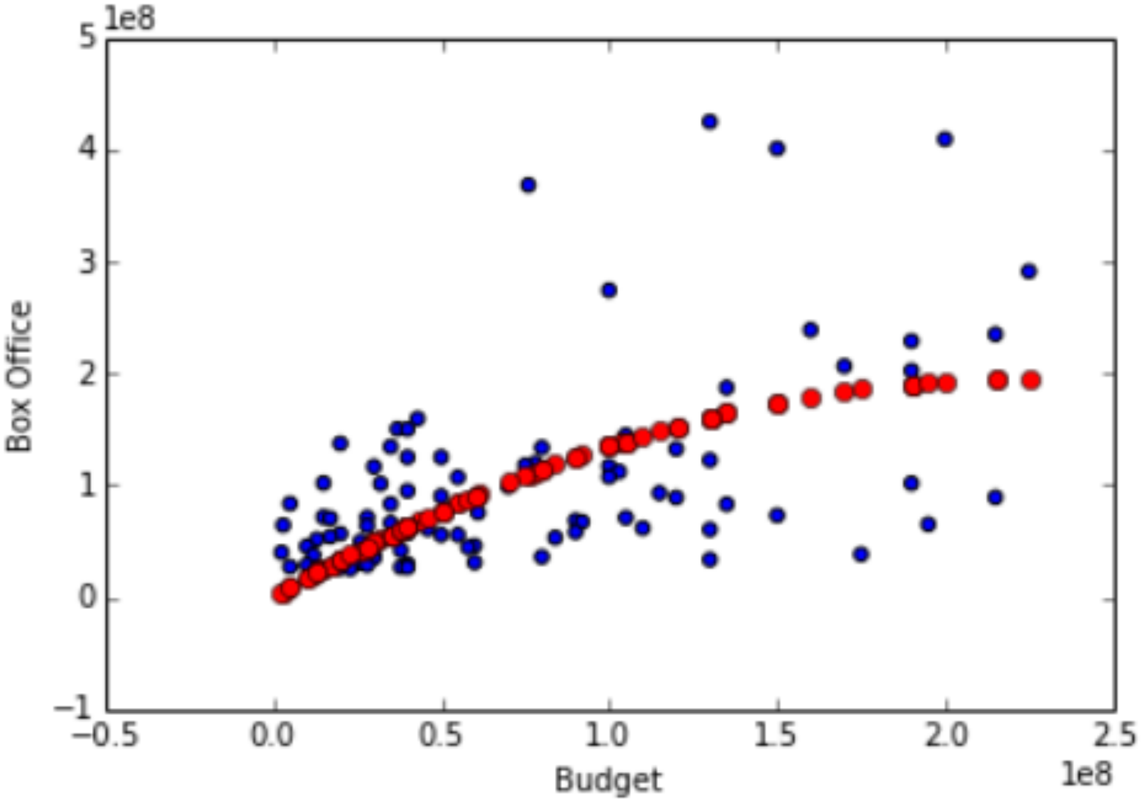
# Polynomial regression

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



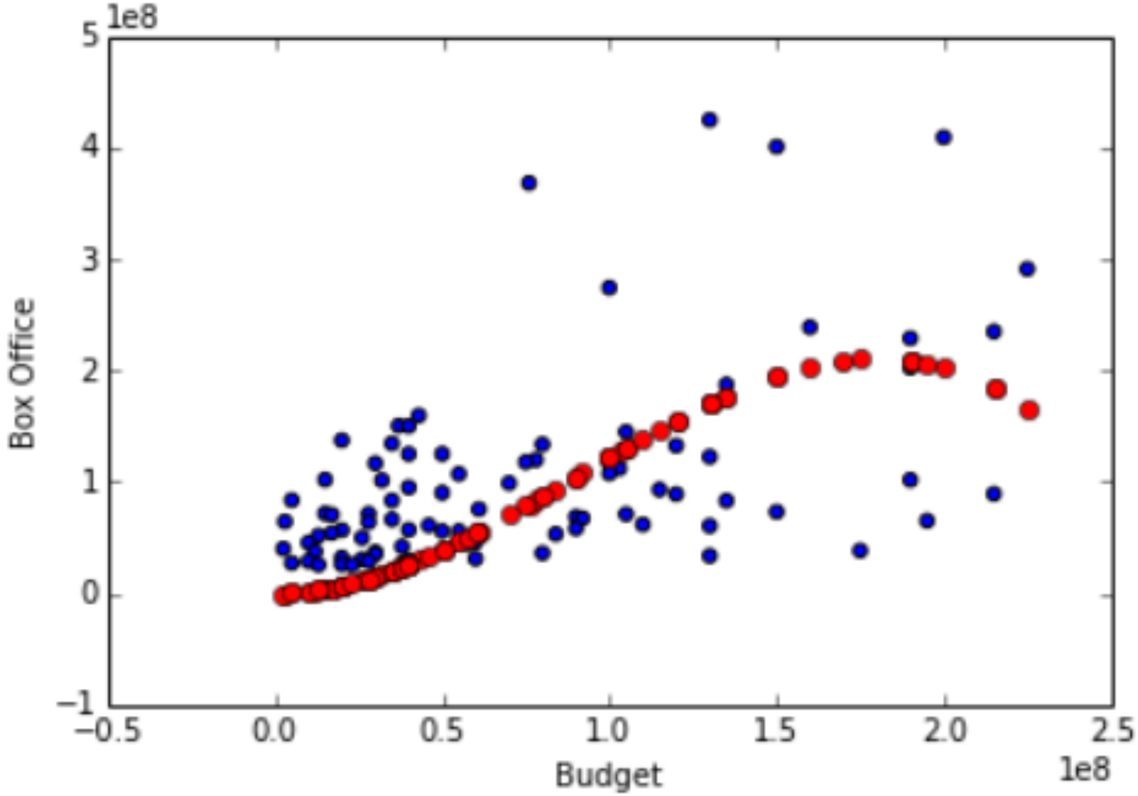
# Polynomial regression

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



# Polynomial regression

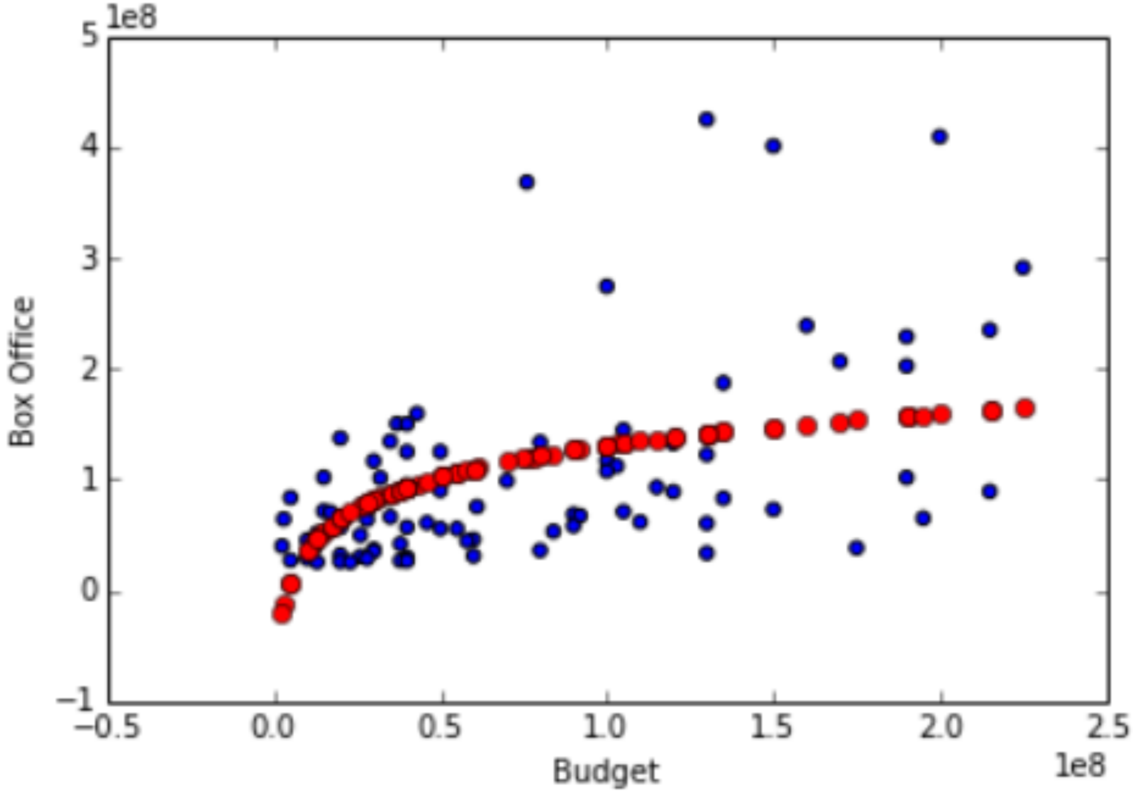
$$y_\beta(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$



# Other functional forms

log

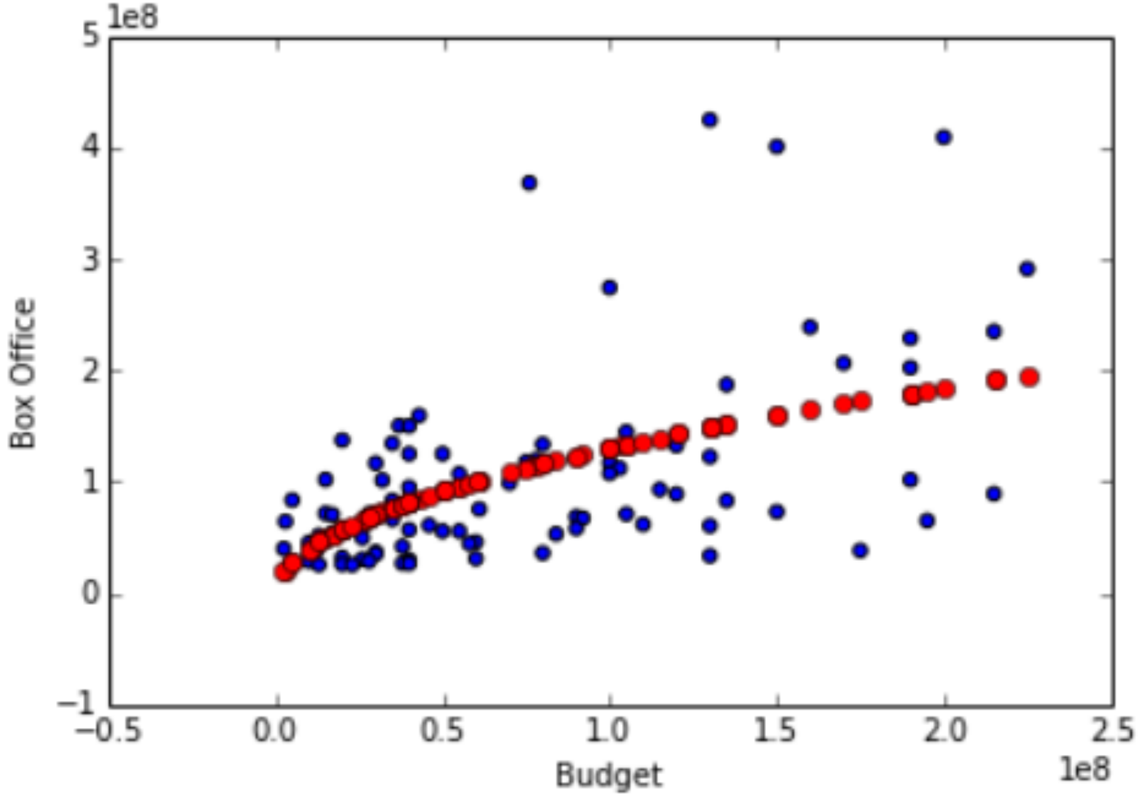
$$y_\beta(x) = \beta_0 + \beta_1 \log(x)$$



# Other functional forms

## square root

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



# Possible to combine variables

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

## Possible to combine variables

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

## Interactions

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

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

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

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

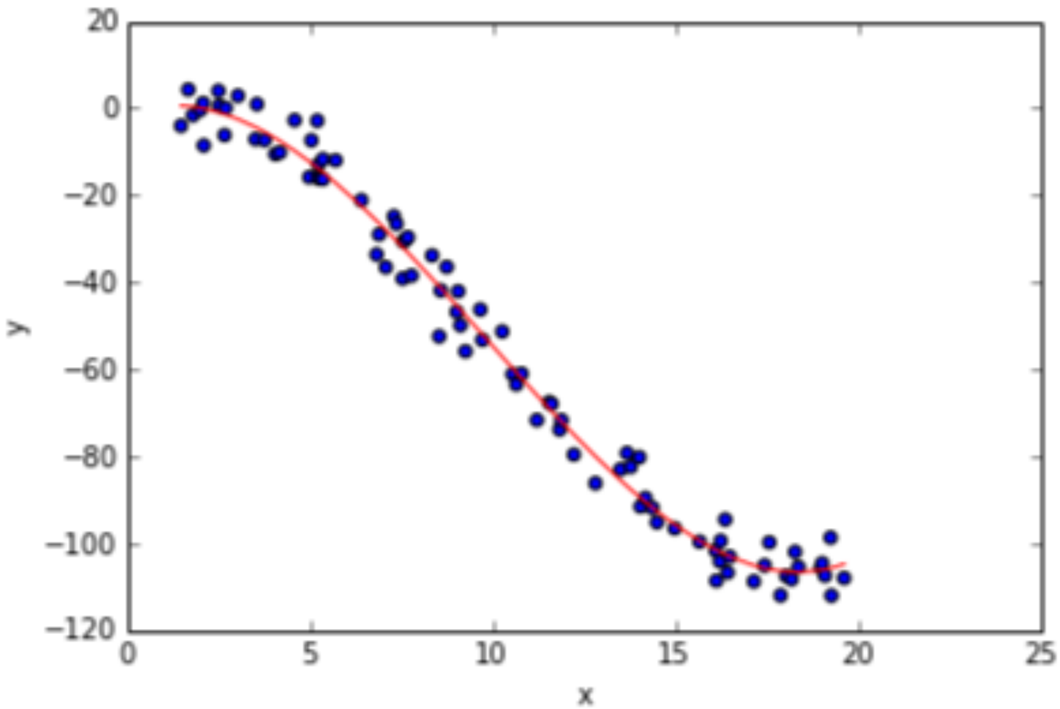
$$y_\beta(x) = \beta_0 + \beta_1 \exp(x_1) + \beta_2 x_2^{-1}$$

Nonlinear

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

# How to choose functional forms to try?

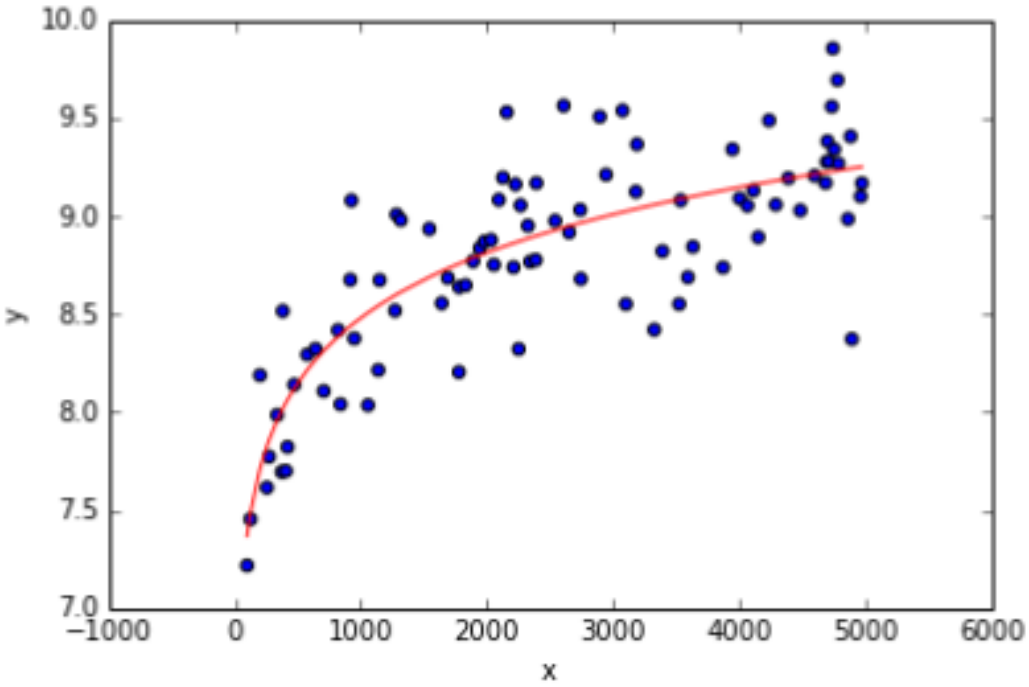
## Check one on one relationship of variable with outcome




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

# How to choose functional forms to try?

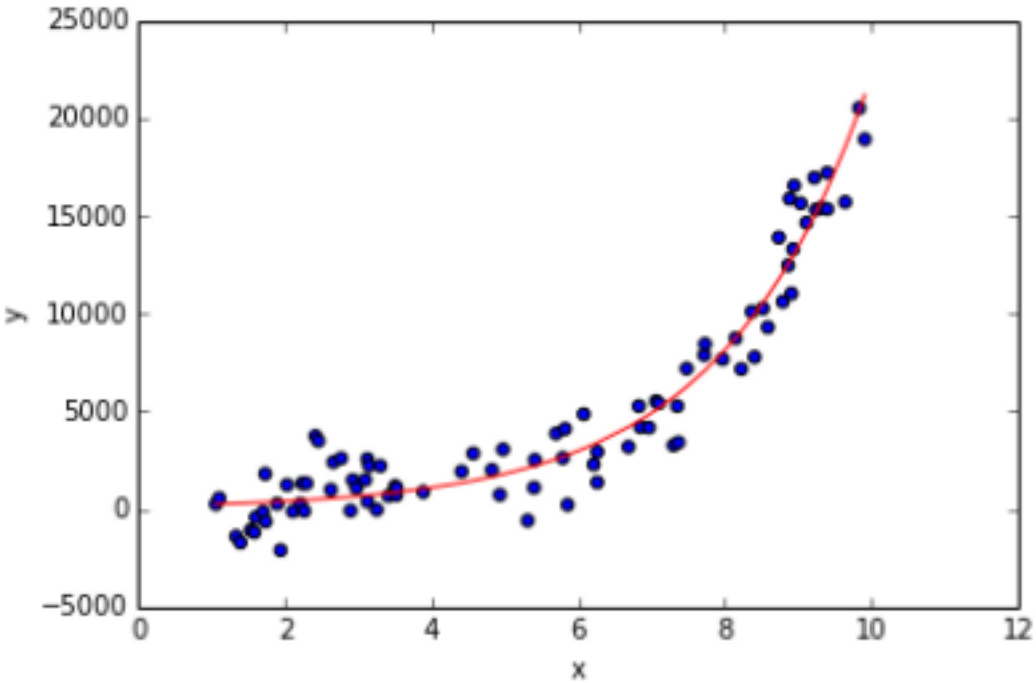
## Check one on one relationship of variable with outcome




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

# How to choose functional forms to try?

## Check one on one relationship of variable with outcome




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

# Data Science Killer #1: Overfitting

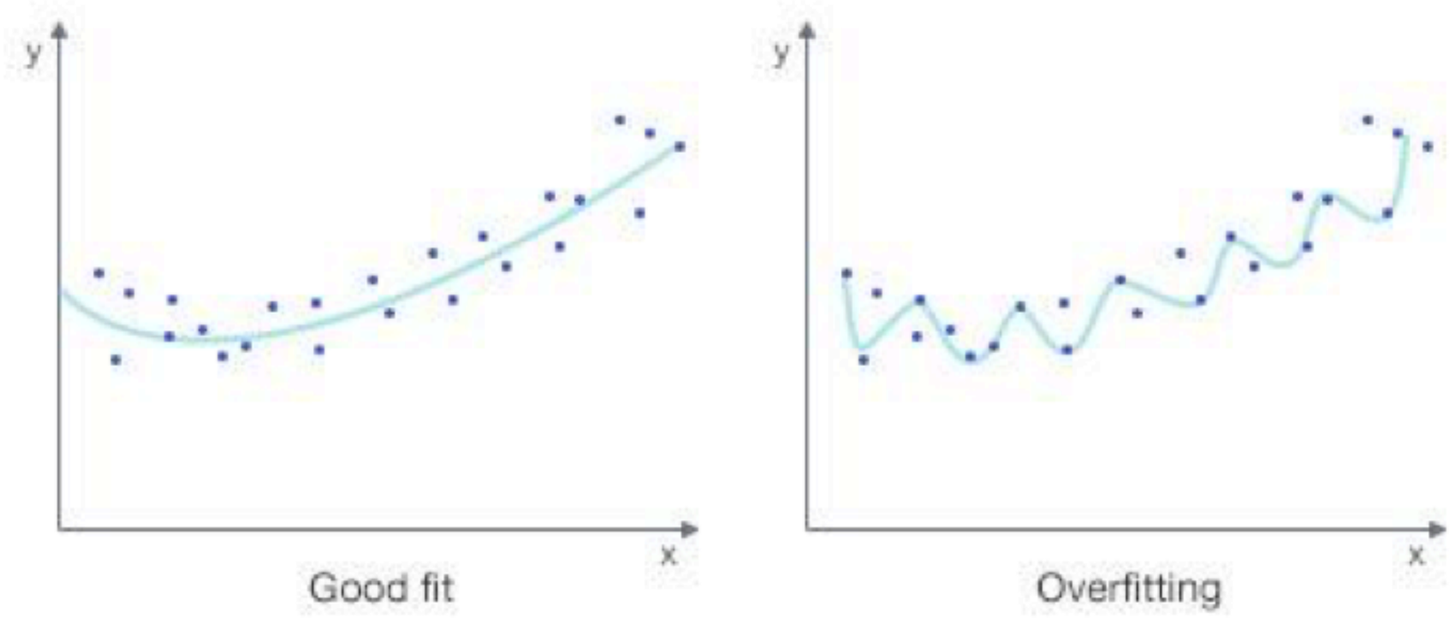
# What is Overfitting?

When I fit too closely to my training set

Why is this bad?

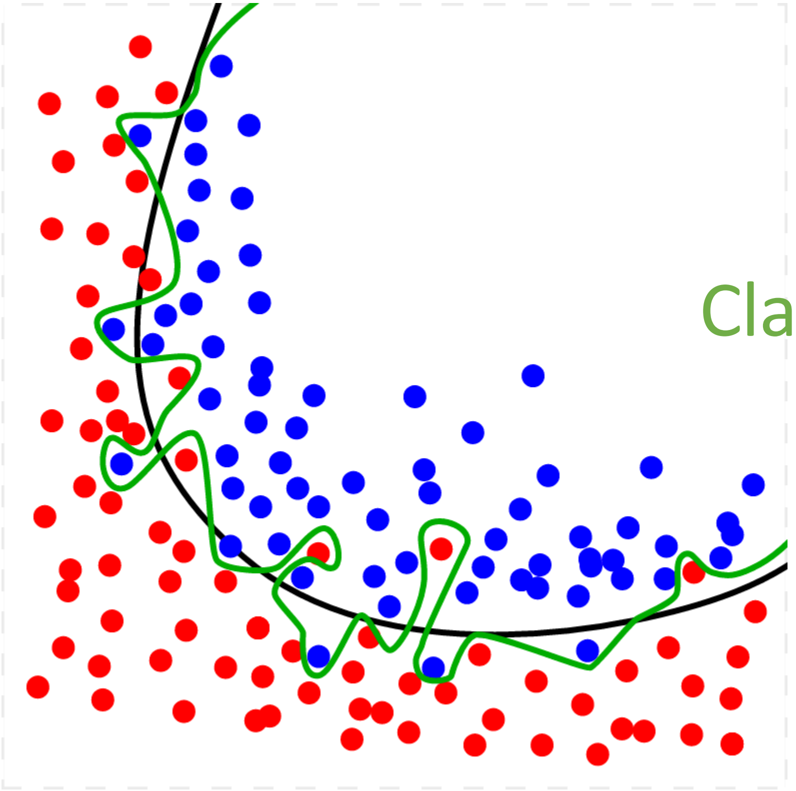
Because my model won’t **generalize** well to **future data**!

# What is Overfitting?




Regression

# What is Overfitting?



Classification

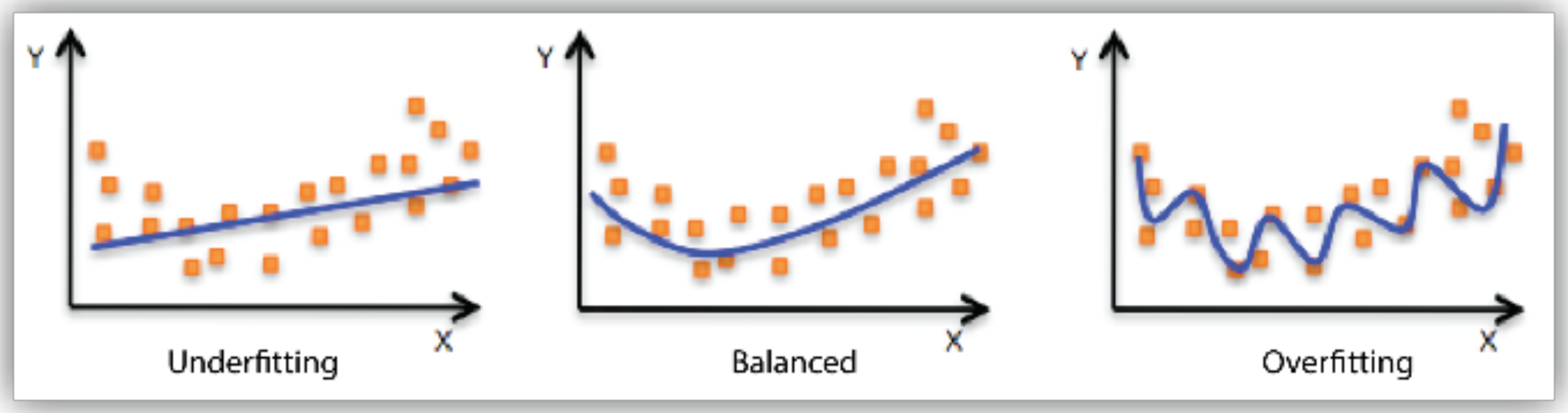
# What is Underfitting?

When I don’t have a complex enough model to model my data.

Why is this bad?

Because we are losing information!

# What is Underfitting/Overfitting?

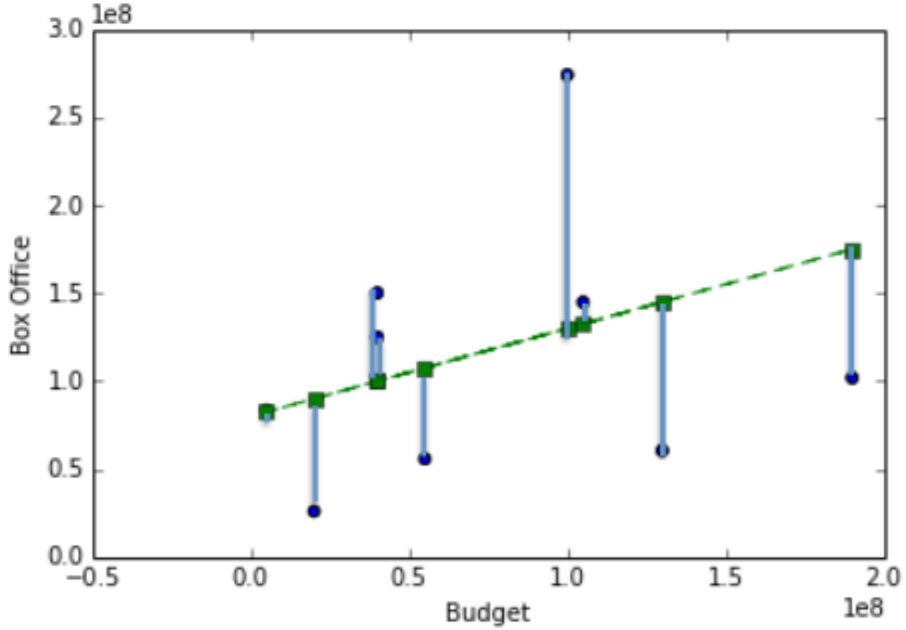



Regression

# Regularization

While awarding goodness of fit, penalize model complexity

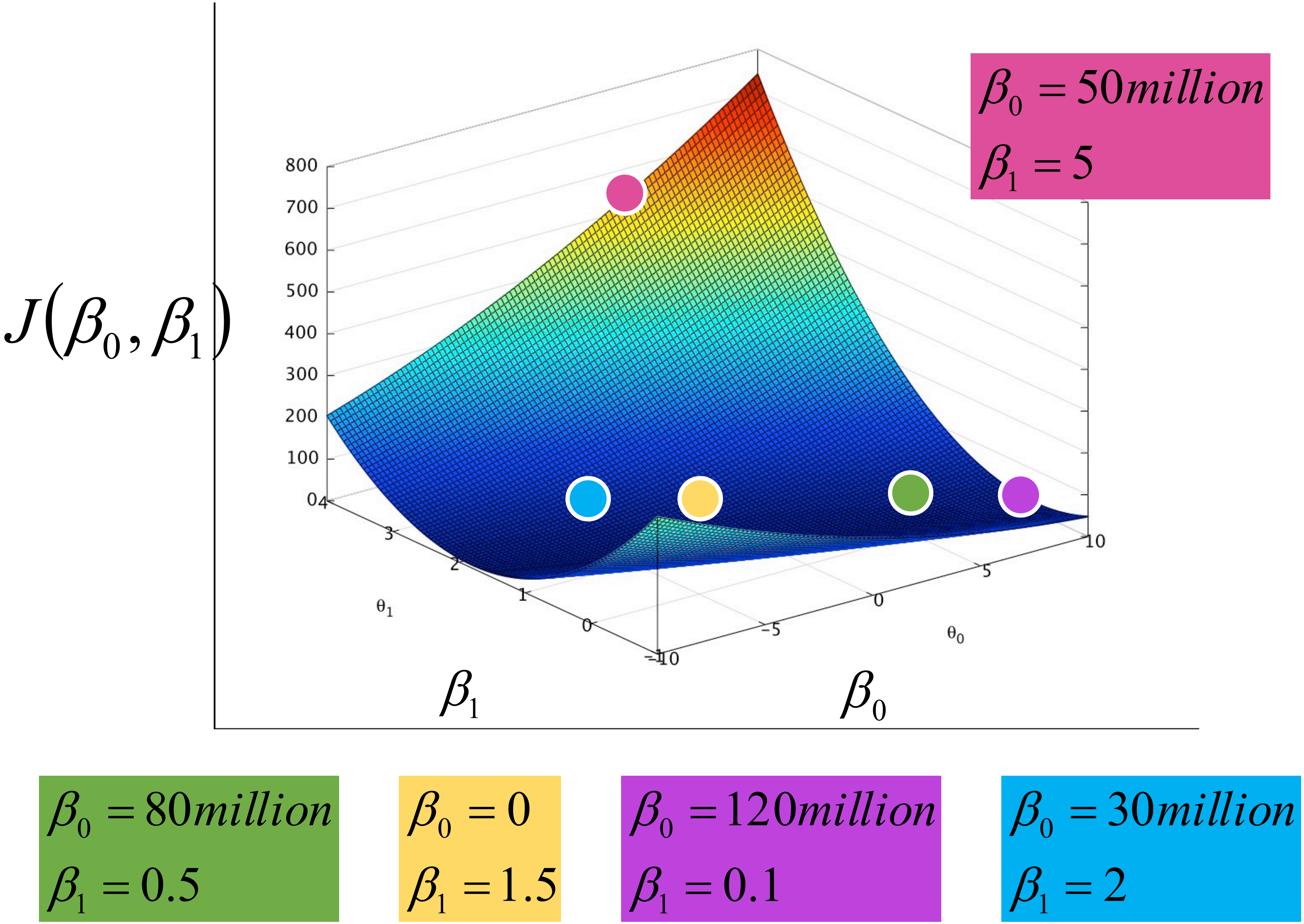
Why not do that while we are fitting?




# Cost function

## Takes a model (specific parameter values), returns a score

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


$J(\beta_0, \beta_1)$
800
700
600
500
400
300
200
100
0
4
3
2
1
0
−10
$\theta_1$
−5
0
5
10
$\theta_0$
$\beta_1$
$\beta_0$
$\beta_0 = 50 million$
$\beta_1 = 5$
$\beta_0 = 80 million$
$\beta_1 = 0.5$
$\beta_0 = 0$
$\beta_1 = 1.5$
$\beta_0 = 120 million$
$\beta_1 = 0.1$
$\beta_0 = 30 million$
$\beta_1 = 2$

# Cost function

$$J(\beta_0, \beta_1) = \frac{1}{2m} \sum_{i=1}^{m} \underbrace{\left( (\beta_0 + \beta_1 x_{obs}^{(i)}) - y_{obs}^{(i)} \right)^2}_{\text{Lower for better fits}}$$

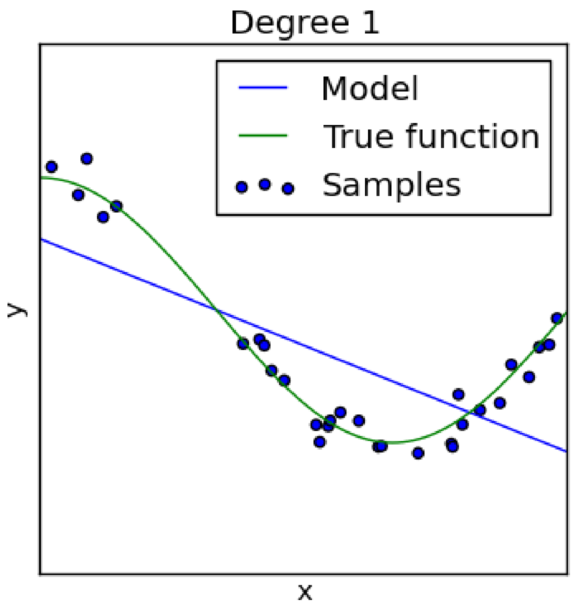
# Cost function

Add a penalty for the size of each parameter!

$$J(\beta_0, \beta_1) = \underbrace{\frac{1}{2m}\sum_{i=1}^{m}\left(y_\beta(x_{obs}^{(i)}) - y_{obs}^{(i)}\right)^2}_{\substack{\text{Low: good fit} \\ \text{High: bad fit}}} + \underbrace{\lambda \sum_{j=1}^{k} \beta_j^2}_{\substack{\text{Low: simple model} \\ \text{High: complex model}}}$$

# Diagnostics to detect under/overfitting

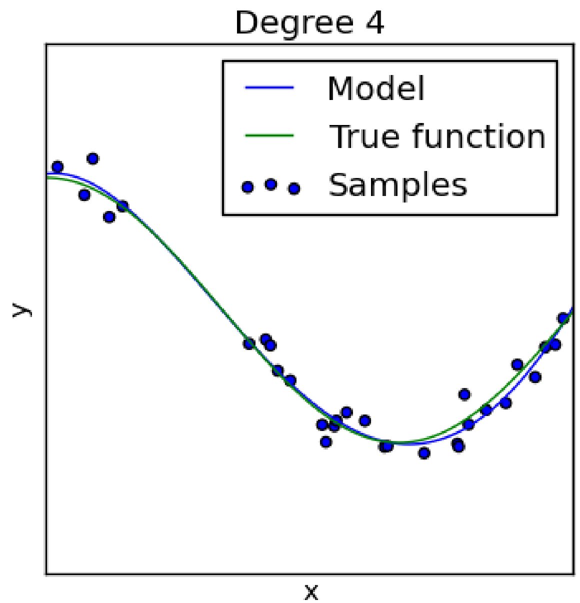
## Underfitting




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

J = V. High + Low

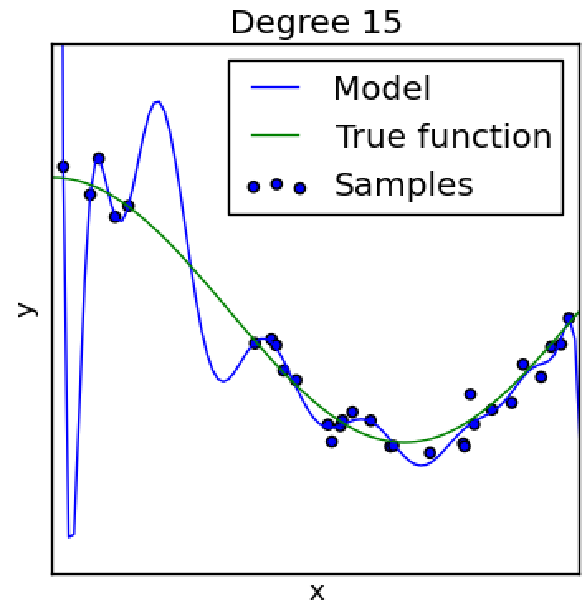
## Just Right




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

J = Low + Low

## Overfitting




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

J = Low + V. High

# Ridge Regression

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

Underfitting

J = V. High + Low

Just Right

J = Low + Low

Overfitting

J = Low + V. High

λ=1

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

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

Underfitting
J = V. High + Low

Just Right
J = Low + Low

Overfitting
J = Low + V. High

λ=1

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

≈0

≈0

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

Underfitting

J = V. High + Medium

Just Right

J = Low + V High

Overfitting

J = Low + VVHigh

VERY LARGE

underfit

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

≈0 ≈0 ≈0 ≈0

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

Underfitting
J = V. High + Tiny

Just Right
J = Low + Tiny

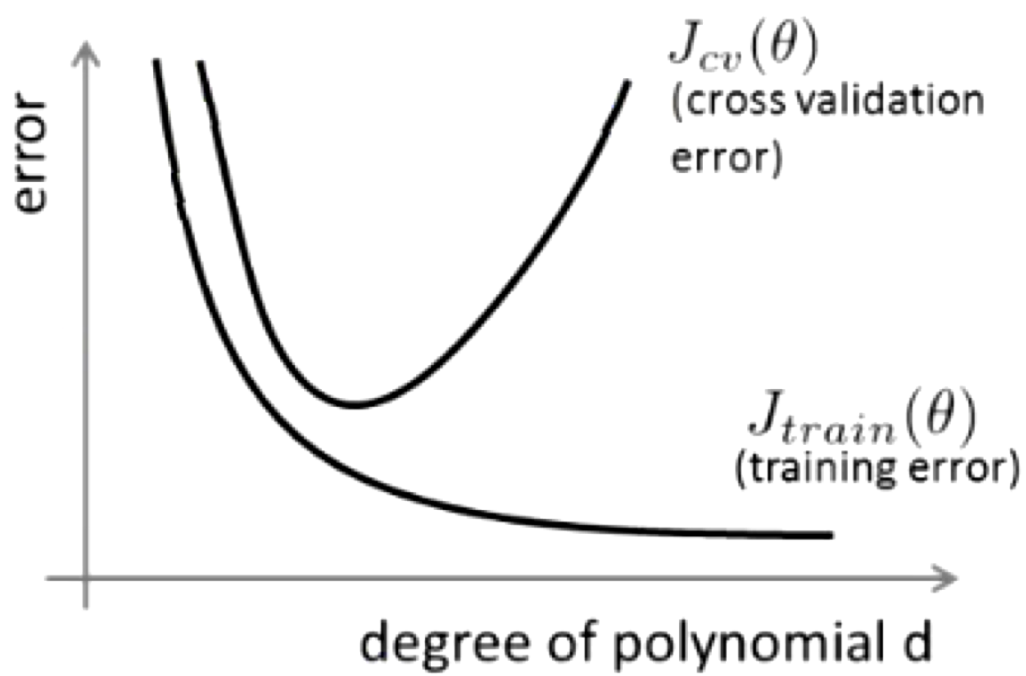
Overfitting
J = Low + Tiny

very small
possible
overfit

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

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

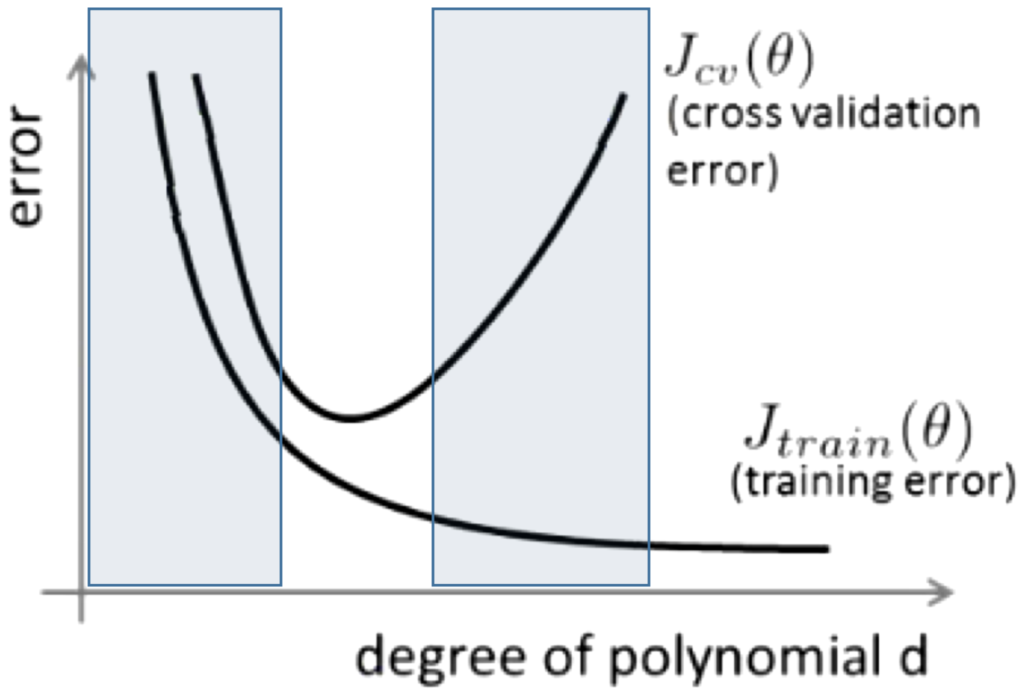
# Error vs. regularization λ




High λ

Low λ

Underfit Overfit




High λ Low λ

# Ridge Regularization (L2)

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

## Ridge Regularization (L2)

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

## Lasso Regularization (L1)

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

## Ridge Regularization (L2)

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

## Lasso Regularization (L1)

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

## Elastic Net (L1 + L2)

$$J(\beta_0, \beta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left( y_\beta (x_{obs}^{(i)}) - y_{obs}^{(i)} \right)^2 + \lambda_1 \sum_{j=1}^{k} \left| \beta_j \right| + \lambda_2 \sum_{j=1}^{k} \beta_j^2$$

My model is not awesome enough.

What do I do?

Try these and check test error
(and AIC,BIC,etc.) again:

Use a smaller set of features

**Regularization: Increase/decrease λ**

Try adding polynomials

Check functional forms for each feature

Try including other features

Use more data (bigger training set)