INTRO TO MACHINE LEARNING AND LINEAR REGRESSION

March 03, 2015 – General Assembly, Santa Monica

Mohsen Chitsaz, Ph.D.
Sr. Principal Data Scientist at Symantec

Lecture 06

Machine Learning

"Field of study that gives computers the ability to learn without being explicitly programmed"

-Arthur Samuel, 1959

"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E"

- Tom M. Mitchell

Types of Variables

- **Continuous (Quantitative)**
  - Interval
  - Ratio

- **Categorical (Qualitative and Discrete)**
  - Nominal
  - Ordinal
  - Flags (Dichotomous)

Types of Variables

- **Continuous (Numerical)**
  - Interval
    - The difference between two measurements matter, <u>scale is not preserved</u>
      - e.g. 40°C - 20°C, compared to 30°C - 10°C
      - 40°C <u>is not four times</u> warmer than 10°C
  - Ratio
    - An interval quantity that also preserves scale
      - e.g. Height, 6ft is twice as much as 3ft
      - Kelvin temperature (°K)
      - Distance

Types of Variables

- **Categorical (Discrete and Qualitative)**
  - Nominal
    - {'blue', 'black', 'yellow'}
  - Ordinal
    - {'High', 'Medium', 'Low'}
  - Flags (Dichotomous)
    - {'Male', 'Female'}
    - {'True', 'False'}
    - {'Approved', 'Denied'}

Machine Learning

- **Supervised**
  - Regression
  - Classification

- **Unsupervised**
  - Clustering
  - Density estimation
  - Dimensionality reduction

Machine Learning Examples

- **Text classification**
- **Image classification**
- **Machine vision**
- **Pattern recognition**
- **Anomaly detection**

Example – Credit Application

age	23 years
gender	male
annual salary	\$30,000
years in residence	1 year
years in job	1 year
current debt	\$15,000
...	...

https://work.caltech.edu/telecourse.html

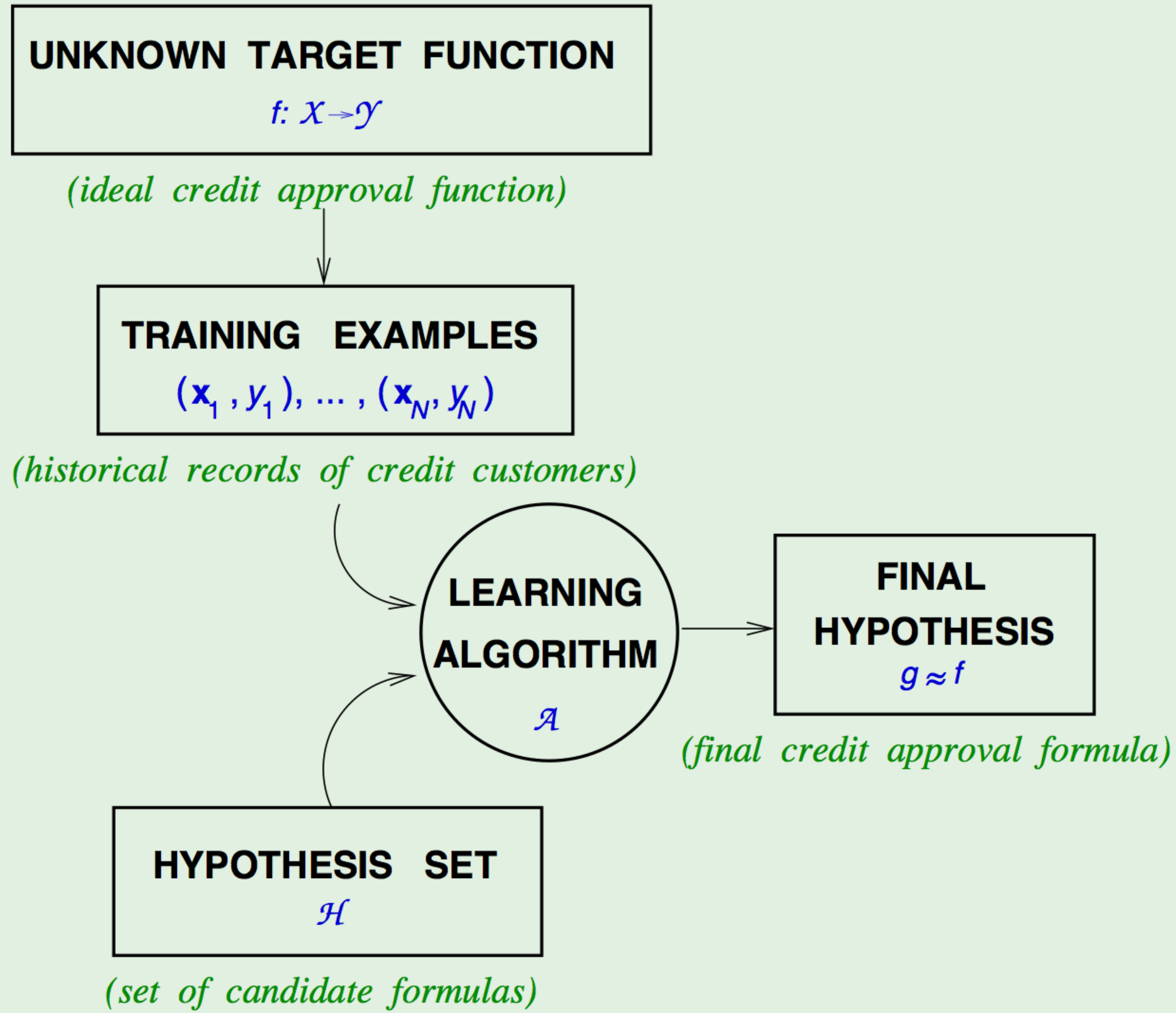
Components of Learning

- Input: $\mathbf{x}$ *(customer application)*
- Output: y *(good/bad customer?)*
- Target function: $f : \mathcal{X} \to \mathcal{Y}$ *(ideal credit approval formula)*
- Data: $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_N, y_N)$ *(historical records)*

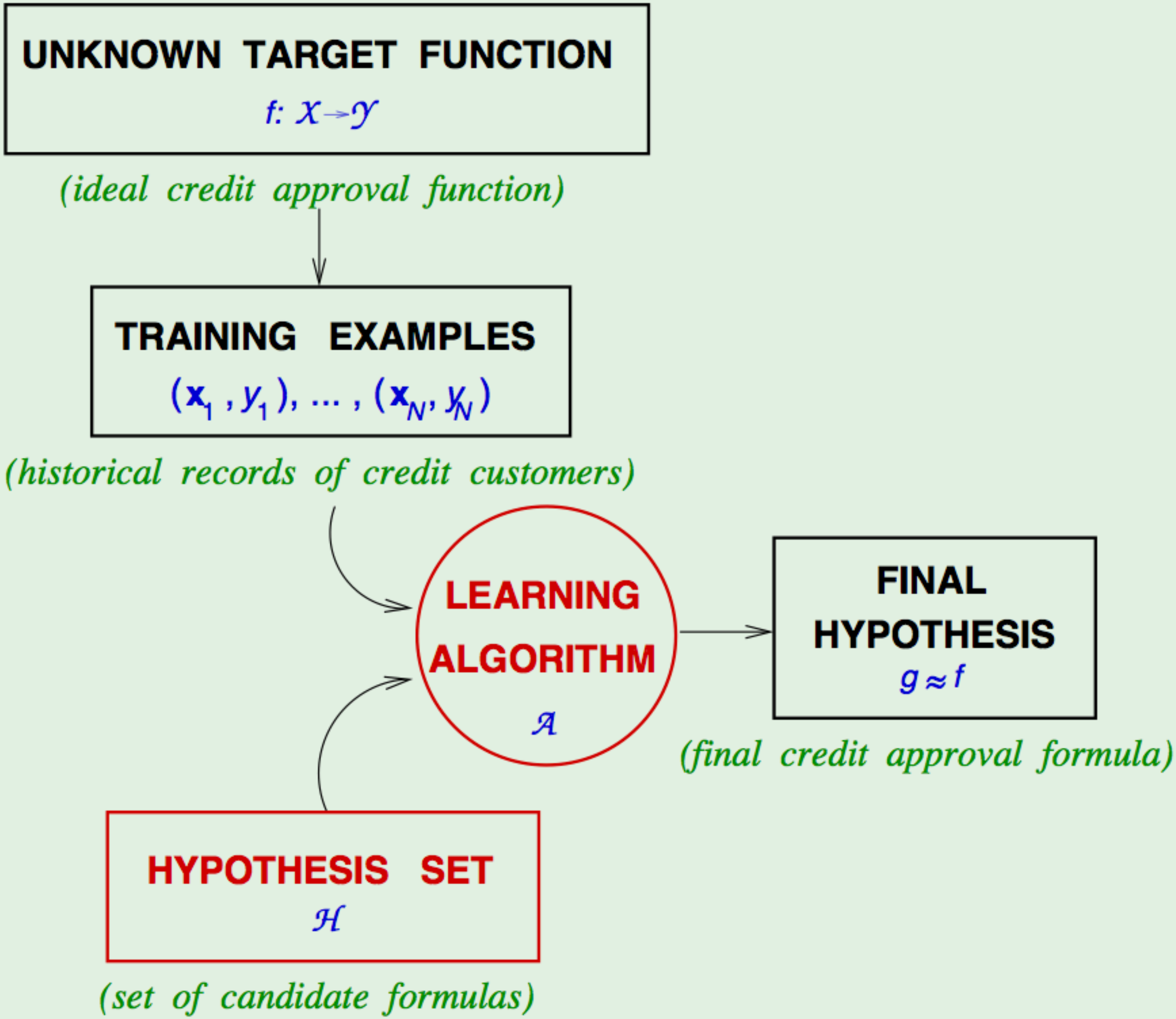
  $\downarrow \quad \downarrow \quad \downarrow$

- Hypothesis: $g : \mathcal{X} \to \mathcal{Y}$ *(formula to be used)*

https://work.caltech.edu/telecourse.html

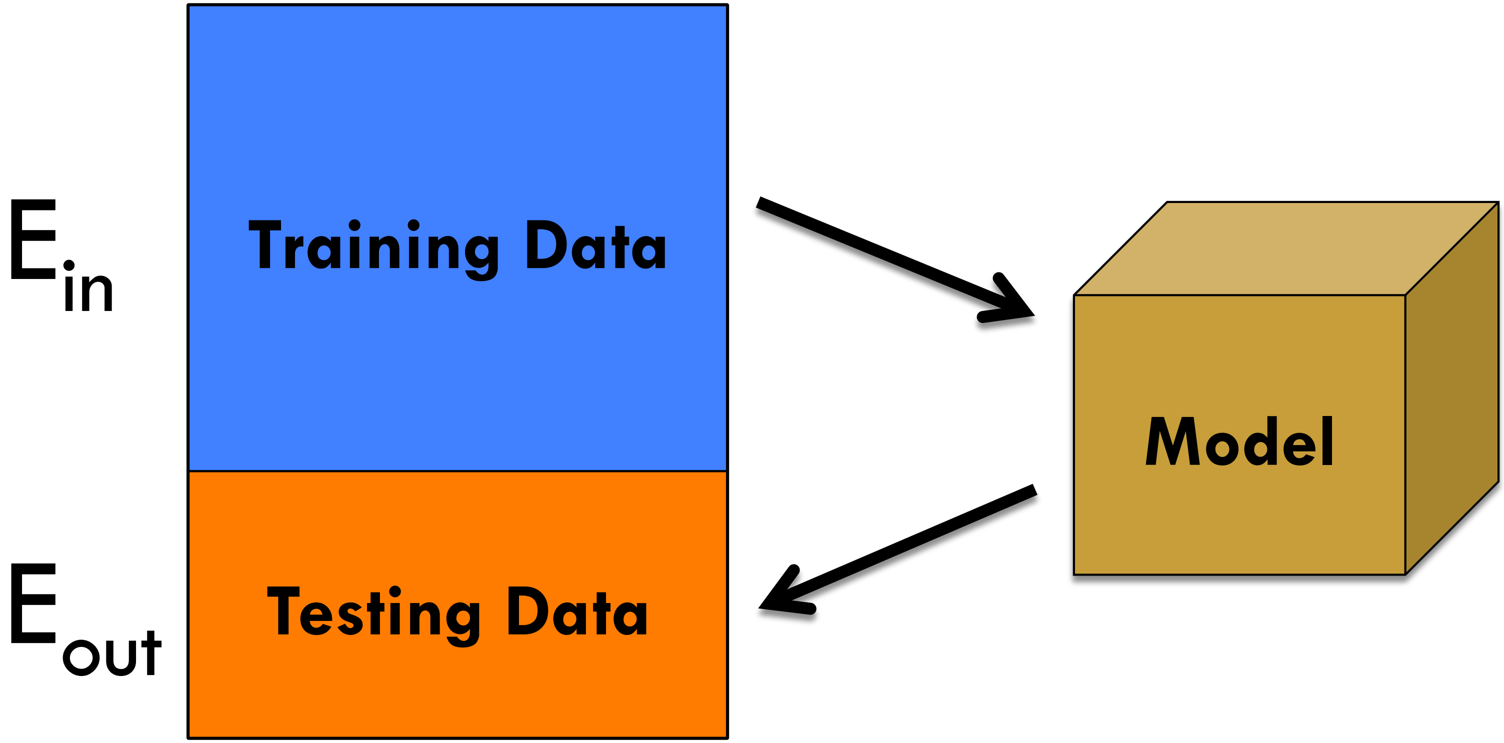


https://work.caltech.edu/telecourse.html



https://work.caltech.edu/telecourse.html

Training, and Testing Data Sets



E_{in}: In sample error

E_{out}: Out of sample error

Learning Tradeoffs (Bias, Variance)

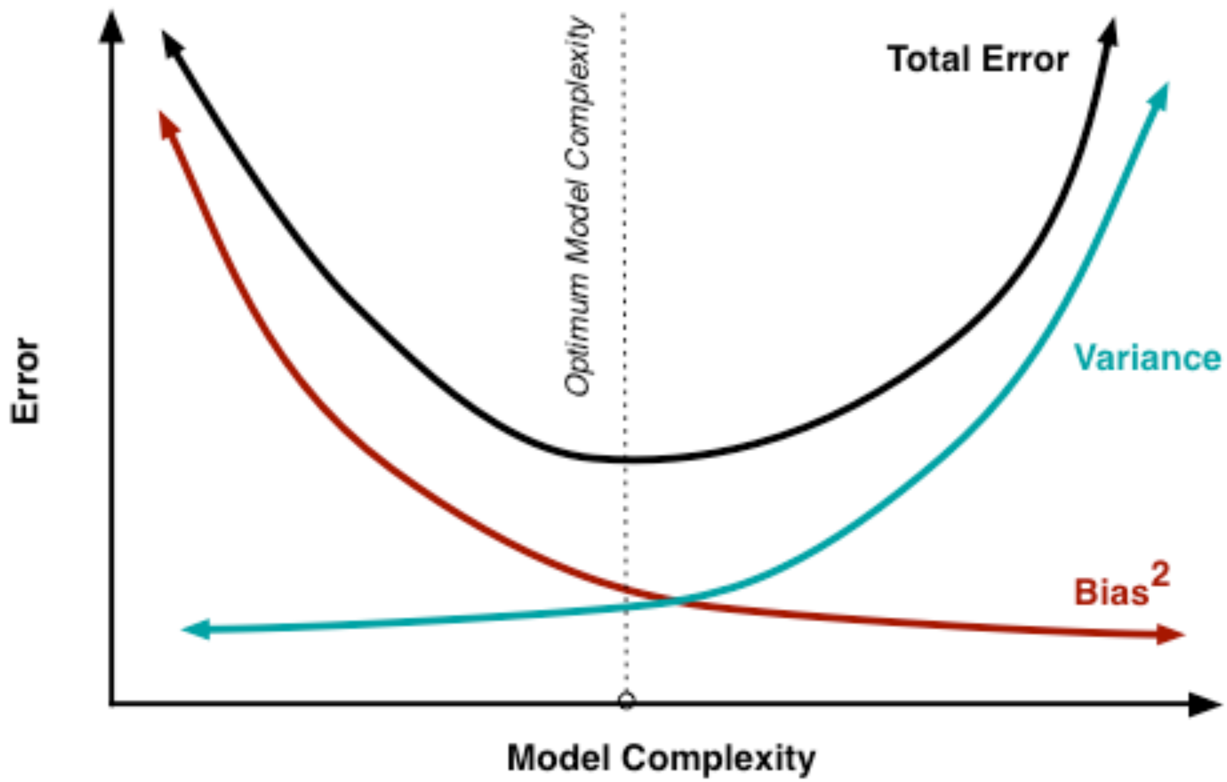
Hypothesis set complexity ↑

- E_{in} (in sample error) <u>decreases</u>
- E_{out} (out of sample error) <u>increases</u>

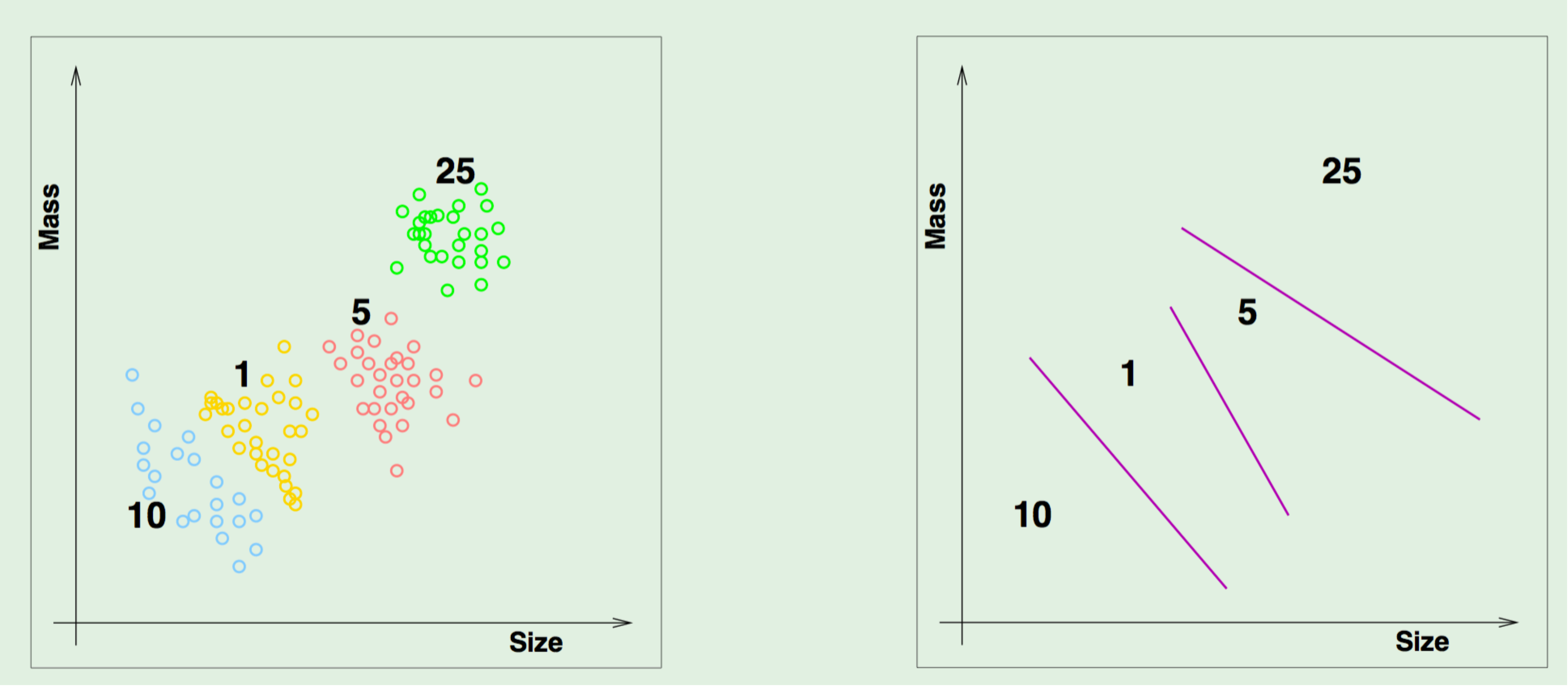
Hypothesis set complexity ↓

- E_{in} (in sample error) <u>increases</u>
- E_{out} (out of sample error) <u>decreases</u>

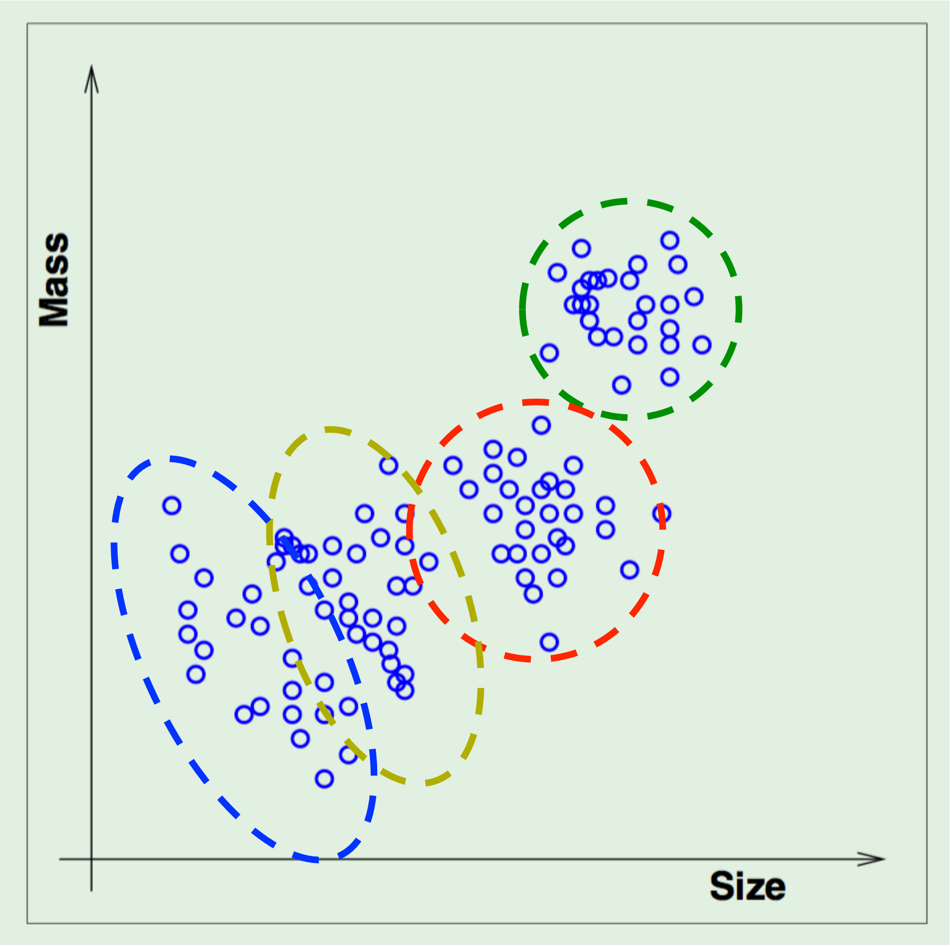
Bias, Variance Tradeoff



Supervised – Coin Recognition



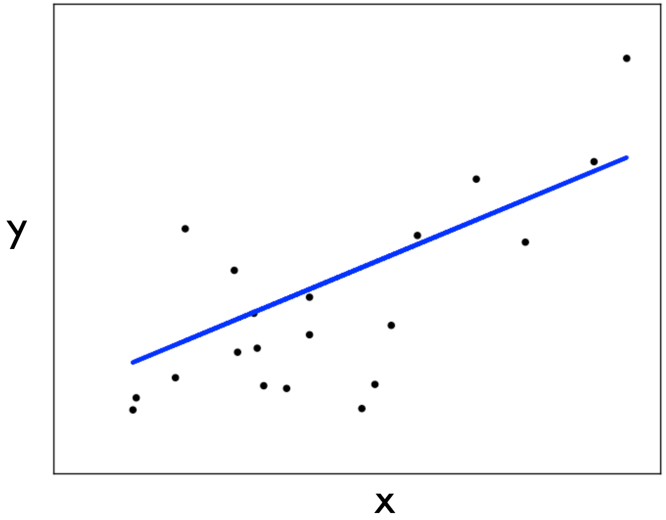
Unsupervised – Coin Recognition



Linear Regression

Generalized Linear Models

$$\hat{y}(w, x) = \underbrace{w_0}_{\text{Intercept}} + \underbrace{w_1 x_1 + ... + w_p x_p}_{\text{weighted inputs}}$$



Ordinary Least Squares

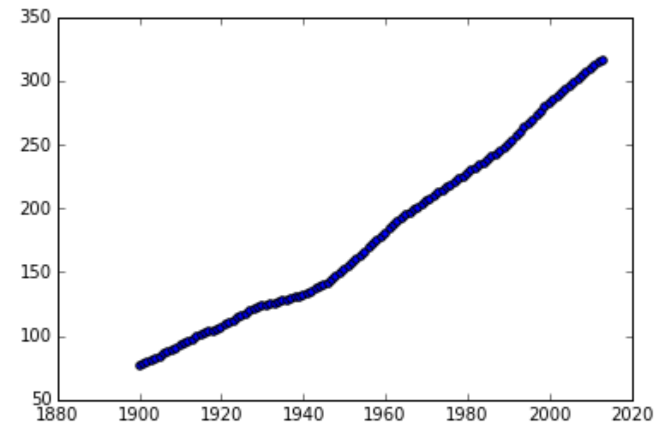
- Minimizes the error,
  X: input vector, **y:** output value, **w:** weights

$$\min_{w} ||Xw - y||_2{}^2$$

- Example:
  - US Population from 1900 - 2013

Regression Example – US Population

Year	Population
1900	76.09
1901	77.58
1902	79.16
1903	80.63
1904	82.17
1905	83.82
1906	85.45
1907	87.01
1908	88.71



US Population Example

```
xl = pd.read_excel('US_Population.xlsx')
```

```
import sklearn.linear_model as linear_model
```

```
model = linear_model.LinearRegression()
Year = xl.Year
Population = xl.Population
```

```
params = model.fit(Population.reshape(Year.size,1), Year)
```

```
print 'coef: ', params.coef_
print 'intercept: ', params.intercept_
```

```
coef:  [ 0.46041748]
intercept:  1873.43878924
```

How good is our model?

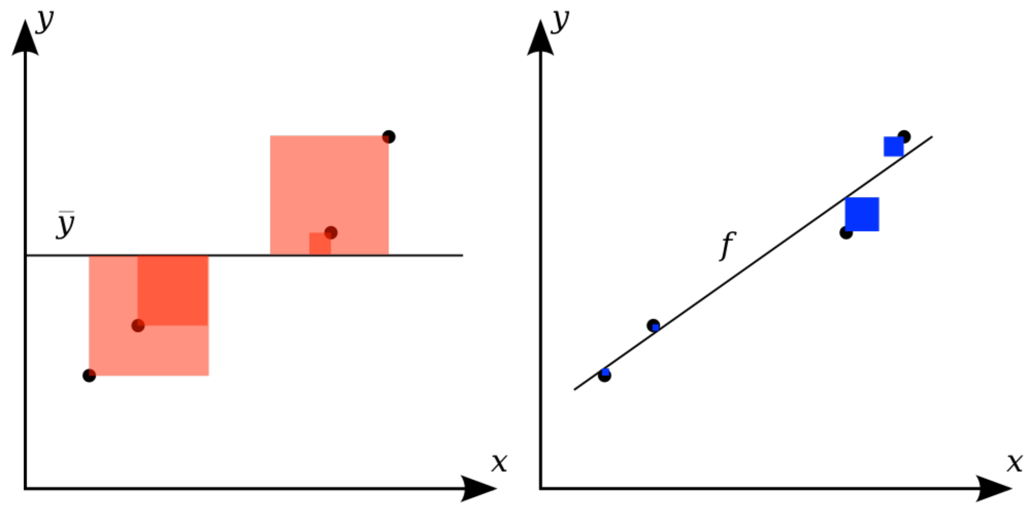
- We need to know how well the model fits the data

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$SS_{\text{tot}} = \sum_i (y_i - \bar{y})^2,$$

$$SS_{\text{res}} = \sum_i (y_i - f_i)^2$$

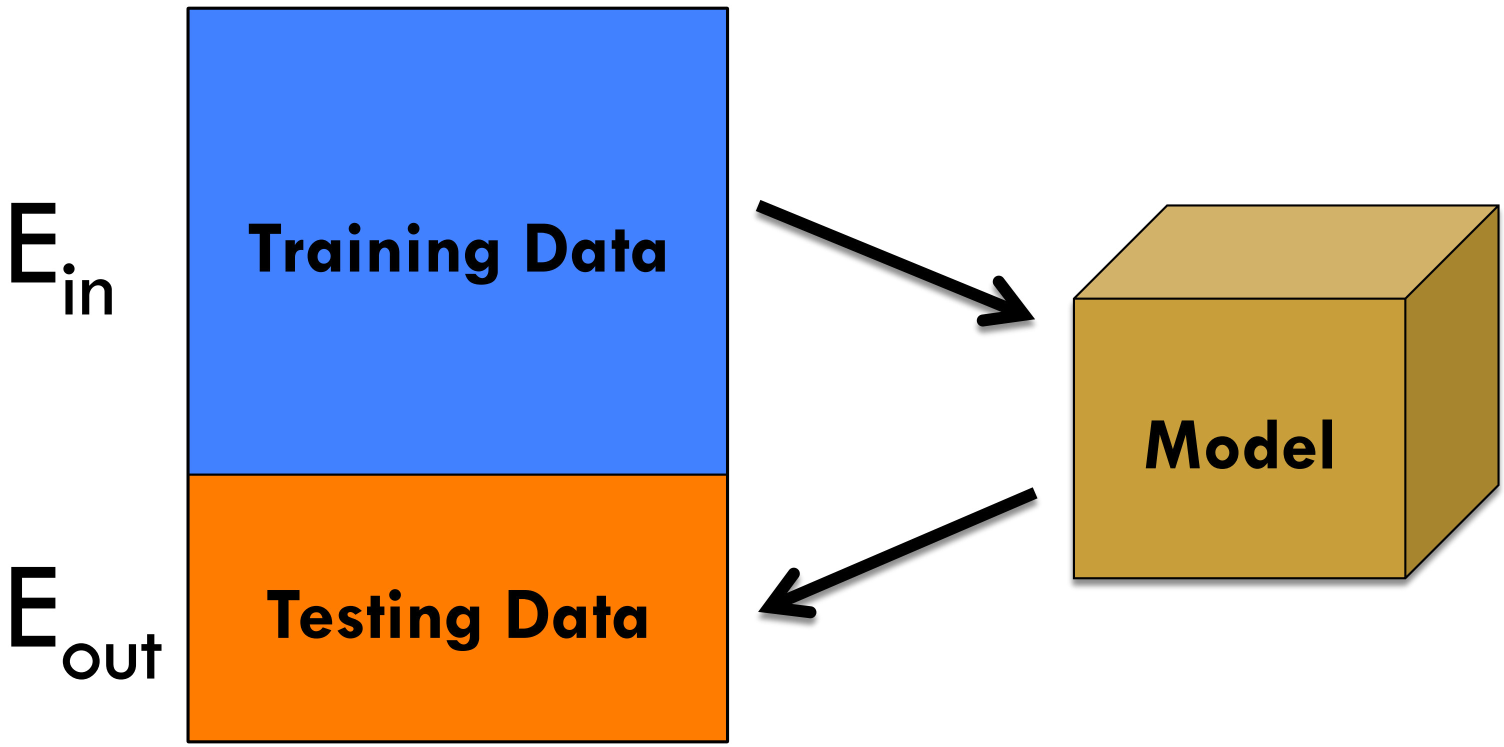
$$R^2 \equiv 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}.$$



`sklearn.metrics.r2_score`

Anything missing in our Regression?

Training, and Testing Data Sets



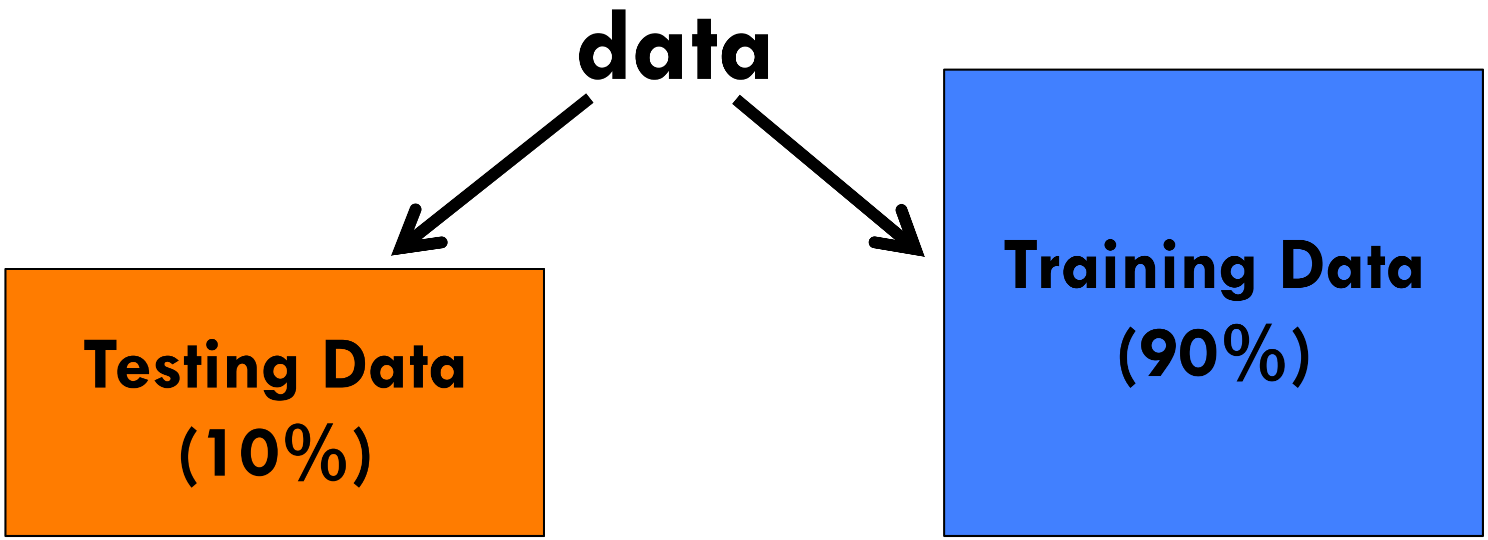
E_{in}: In sample error

E_{out}: Out of sample error

US Population Example

```
import sklearn.cross_validation as cv
```

```
split_ratio = 0.10
split_data = cv.train_test_split(xl, test_size=split_ratio)
training_data = split_data[0]
testing_data = split_data[1]
```



Youtube Rating Prediction

1) Search for videos "*maddona*"

2) *Parse the json result*

3) *Extract features of videos, e.g. number of likes, number of dislikes, number of views, number of days since its publish date*

4) *Come up with a formula that relates features to the rating*

https://gdata.youtube.com/feeds/api/videos?q=hollywood&max-results=50&v=2&alt=json

Class Group Work – Abalone Age

- Determining the age Abalone is very laborious
- We want to find a formula that predicts the **age** of abalone based on some of its features

Class Group Work – Abalone Age

Feature	Description
sex	M, F, I, (Gender or Infant)
length	Longest shell measurement (mm)
diameter	Perpendicular to the length (mm)
height	With meat in shell (mm)
whole_weight (gr)	Whole weight (gr)
shucked_weight	Weight of meat (gr)
viscera_weight	Gut weight after bleeding (gr)
shell_weight	After being dried (gr)
rings	+1.5 gives the age in years

Ridge Regression

- Imposes a penalty on the size of coefficients
- Minimizes a penalized residual sum of squares:

$$\min_{w} ||Xw - y||_2{}^2 + \alpha ||w||_2{}^2$$

- Alpha is a complexity parameter, the greater the alpha is, coefficients become more robust to collinearity

```
model = linear_model.Ridge(alpha = .5)
```

Example

- Try Ridge regression on youtube data and abalone examples
- How does the result of regression changes as we increase alpha?
- Can you plot the out of sample R^2 accuracy as a function of alpha for both problems?