DATA MINING

/CISC 873 - Steven Ding
/Week #1/Lecture 2
Linear Regression & Empirical Error Estimation

Boston Housing Got A PROBLEM

https://www.reddit.com/r/boston/comments/6io6eq/boston_real_estate/

abeuscher 1.2k points · 3 years ago

Bay area: Hold my beer.

1.2 Million in the Bay Area (2017)

https://imgur.com/gallery/aw6hi

Linear Regression – Tabular Data

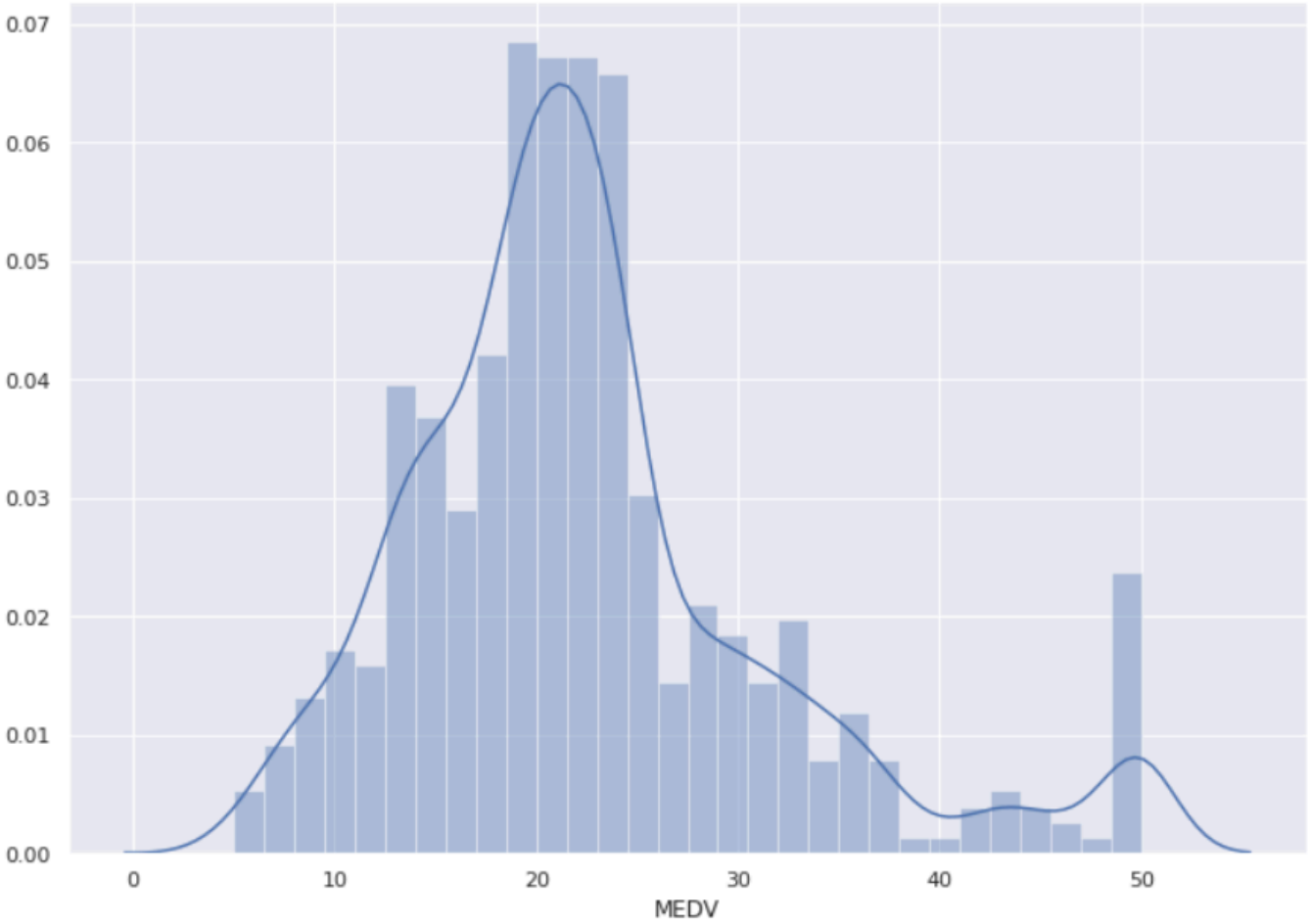
- Boston Housing Data
  - CRIM per capita crime rate by town
  - ZN proportion of residential land zoned…
  - INDUS proportion of non-retail business acres per town
  - NOX nitric oxides concentration (parts per 10 million)
  - RM average number of rooms per dwelling
  - AGE proportion of owner-occupied units built prior to 1940
  - DIS weighted distances to five Boston employment centres
  - …
  - MEDV Median value of owner-occupied homes in $1000's

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Belsley, Kuh & Welsch, 'Regression diagnostics: Identifying Influential Data and Sources of Collinearity', Wiley, 1980. 244-261.

MEDV

sns.distplot(boston['MEDV'], bins=30)



Correlation

sns.heatmap(data=boston.corr().round(2), annot=True)

RM: 0.7

Average number of rooms per dwelling

LSTAT: -0.74

% lower status of the population

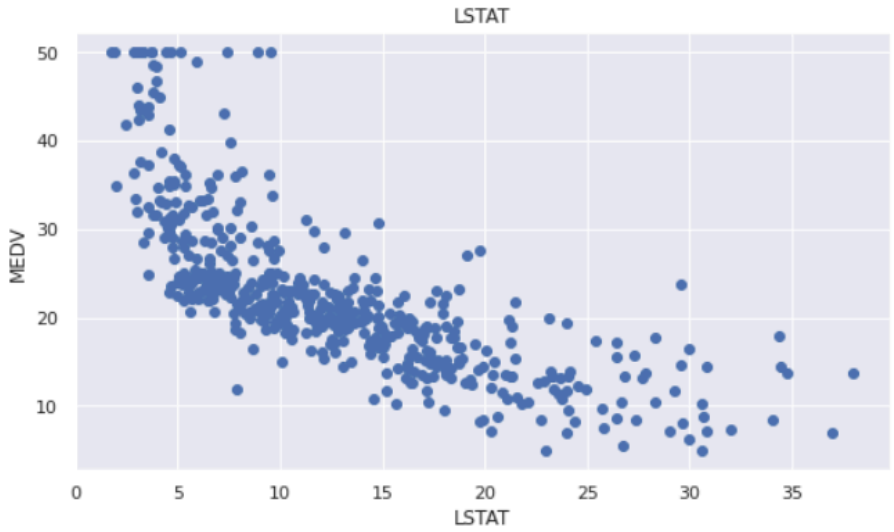
Single-view:

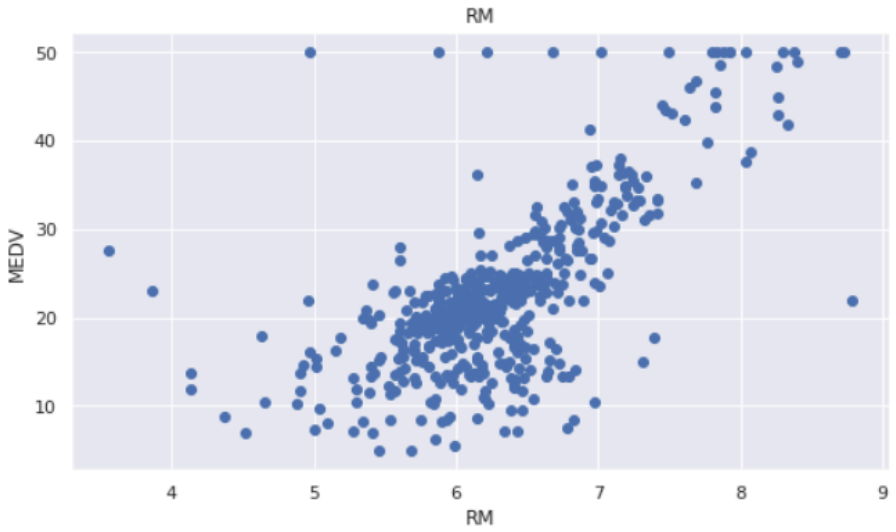
RM: 0.7

Average number of rooms per dwelling

LSTAT: -0.74

% lower status of the population





Supervised Learning

- Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
- New data is classified by the model constructed from the training set

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Supervised Learning

- Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
- New data is classified by the model constructed from the training set

Features

Target Attribute

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Supervised Learning

- Given a dataset of samples for training:

$$x_i =< x_{i,1}, x_{i,2}, x_{i3}, ..., x_{i,m}, y_i >$$

- $x_{i,j}$ is the value of the jth feature for the ith example.
- y_i is the value of the target attribute for the ith example.

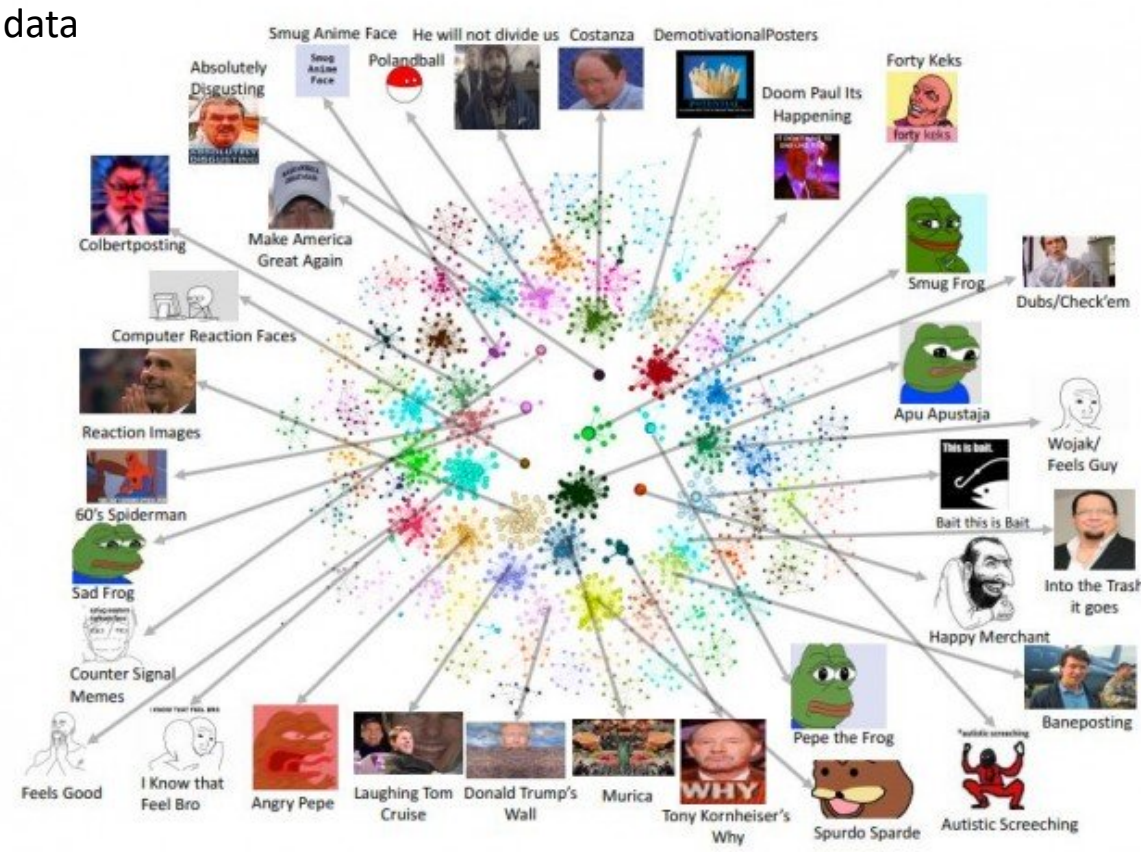
- Goal – Learn a model (function) maps features to target (hypothesis)

$$f : \mathbb{R}^m \rightarrow \mathbb{R}$$

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Other paradigms:

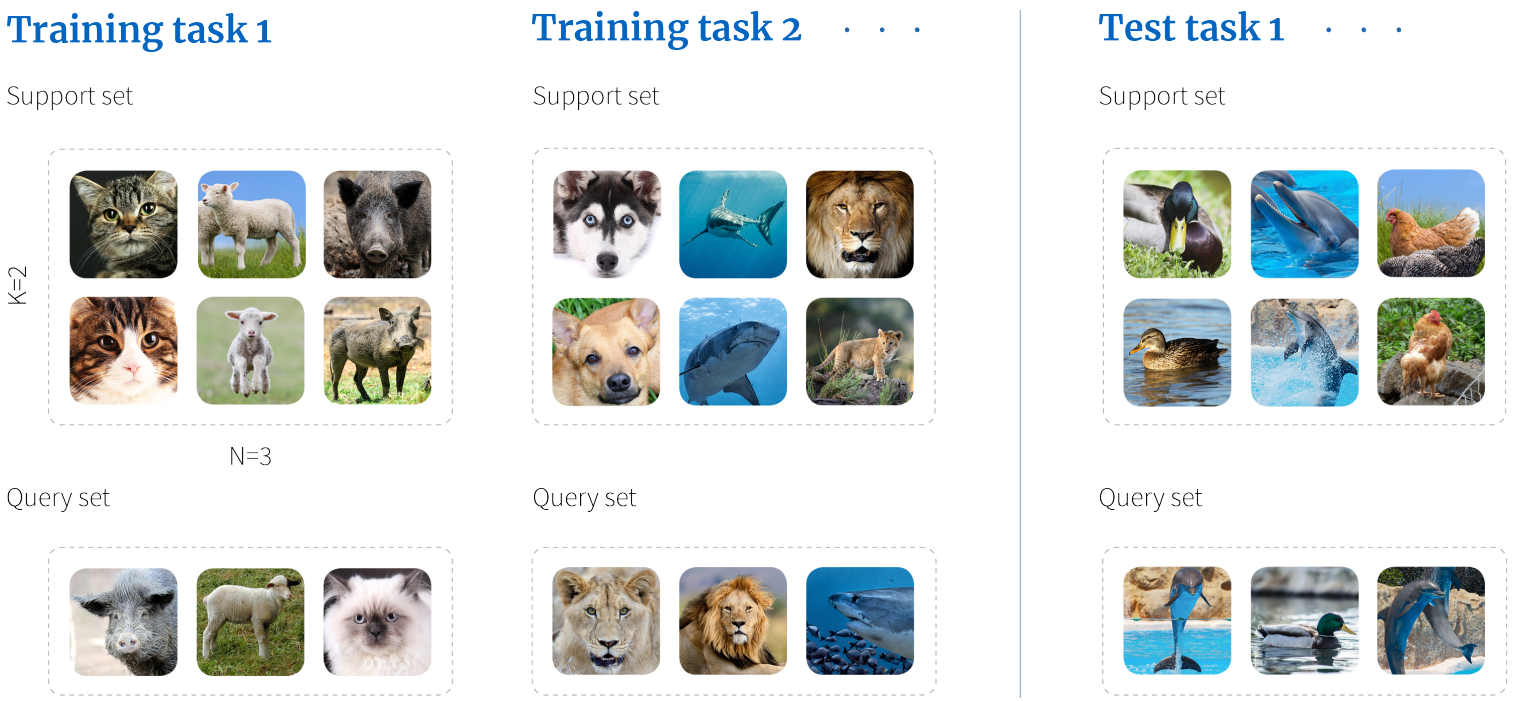
- Unsupervised learning (clustering, representation learning, graphical models)
  - The class labels (directly related to the task/problem) of training data is unknown
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data



https://www.technologyreview.com/2018/06/11/142394/this-is-where-internet-memes-come-from/

Other paradigms:

- Semi-supervised Learning
  - AKA. Weakly supervised
  - A small amount of labeled data with a large amount of unlabeled data during training.
  - Active Learning – Human in the loop
  - Few shot learning, Meta learning, N-way-K-shot classification



Supervised Learning

- Given a dataset of samples for training:

$$x_i =< x_{i,1}, x_{i,2}, x_{i3}, ..., x_{i,m}, y_i >$$

- $x_{i,j}$ is the value of the jth feature for the ith example.
- y_i is the value of the target attribute for the ith example.

- Goal – Learn a model (function) maps features to target (hypothesis)

$$f : \mathbb{R}^m \rightarrow \mathbb{R}$$

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Data Types

$x_i =< x_{i,1}, x_{i,2}, x_{i3}, ..., x_{i,m}, y_i >$

- $x_{i,j}$ is the value of the jth feature for the ith example.
- y_i is the value of the target attribute for the ith example.

- Categorical
  - Nominal: discrete set of size more than 2
    - Hair color = {auburn, black, blond, brown, grey, red, white}
    - marital status, occupation
  - Binary (binominal): discrete set of size 2
    - Symmetric: equally important (gender)
    - Asymmetric: not equally important (HIV positives vs negative)
- Ordinal
  - Ordered discrete set with ranking but no metrics
    - Size = {small, medium, large}, grades, army rankings
  - magnitude between successive values is not known (assumed)
- Numeric
  - Real number measurements can only be measured and represented using a finite number of digits

Classification? Prediction?

$x_i =< x_{i,1}, x_{i,2}, x_{i3}, ..., x_{i,m}, y_i >$

- $x_{i,j}$ is the value of the jth feature for the ith example.
- y_i is the value of the target attribute for the ith example.

- In DM/DS:
  - Classification
    - Target attribute is *categorical*
  - Prediction
    - Target attribute is *numeric*
- In ML:
  - Prediction – Supervised Learning
    - Classification
      - Target attribute is *categorical*
    - Regression
      - Target attribute is *numeric*

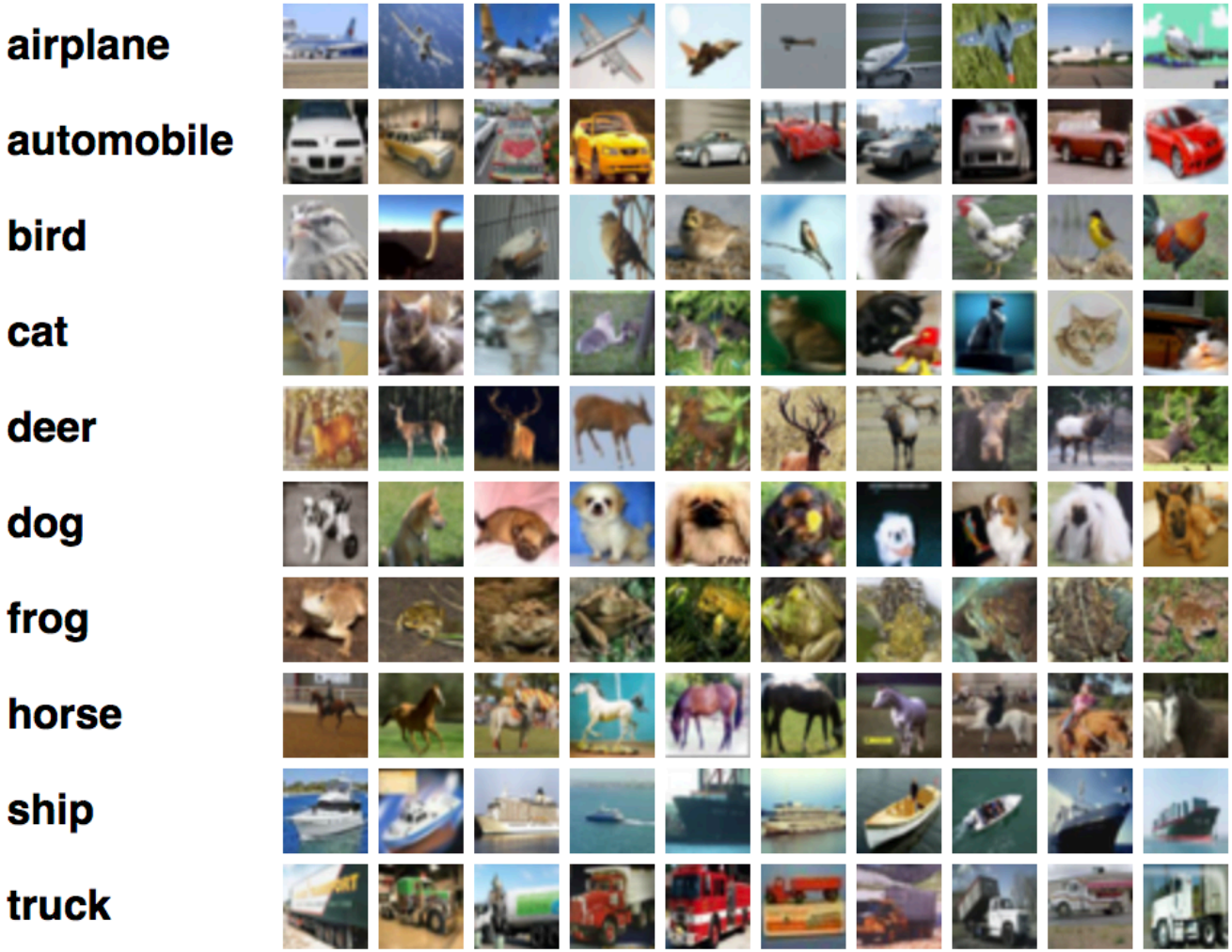
Supervised Learning – i.i.d.

$x_i =< x_{i,1}, x_{i,2}, x_{i3}, ..., x_{i,m}, y_i >$

- $x_{i,j}$ is the value of the jth feature for the ith example.
- y_i is the value of the target attribute for the ith example.

- Each sample x in the dataset are assumed to be independently and identically distributed.
  - Independently: every sample is independently observed from the target system (over domain X of some distributions)
  - Identically: the target system (some distributions) is assumed to be the same for all observations

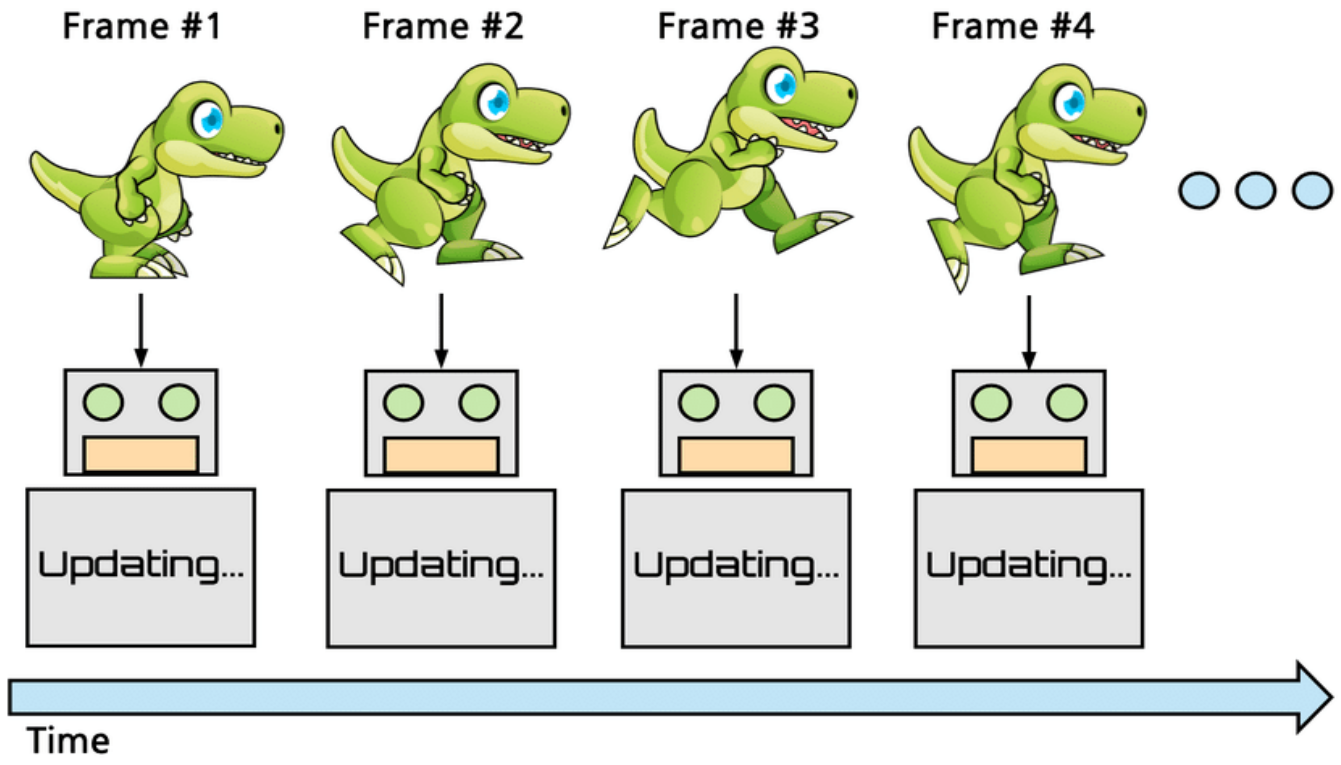
Supervised Learning – i.i.d.



https://machinelearningmastery.com/applications-of-deep-learning-for-computer-vision/

Supervised Learning – non-i.i.d.

- Dependencies exists between observations
  - Sequential learning problem



Maltoni, D., & Lomonaco, V. (2019). Continuous learning in single-incremental-task scenarios. *Neural Networks*, *116*, 56-73.

Supervised Learning

- Given a dataset of samples for training:

$$x_i =< x_{i,1}, x_{i,2}, x_{i3}, ..., x_{i,m}, y_i >$$

- $x_{i,j}$ is the value of the jth feature for the ith example.
- y_i is the value of the target attribute for the ith example.

- Goal – Learn a model (function) maps features to target (hypothesis)

$$f : \mathbb{R}^m \rightarrow \mathbb{R}$$

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

The notion of Error/Risk

- Given our dataset ***D*** and a function class F:
  - The search space of all possible function f such that

$$f : \mathbb{R}^m \rightarrow \mathbb{R}$$

- Define a notion of error:

$$L(f, D)$$

- which measures the errors/mistakes/deviations made on our dataset ***D.***
- Data Mining – Need driven.
- The notion of error is defined based on the problem or the context.

Empirical Risk Minimization

- Given our dataset ***D*** and a function class F:
  - The search space of all possible function f such that

$$f : \mathbb{R}^m \rightarrow \mathbb{R}$$

- Define a notion of error:

$$L(f, D)$$

- ERM: find the best function f in the function class F, such that

$$ERM(D, F) = \text{argmin}_{f \in F} L(f, D)$$

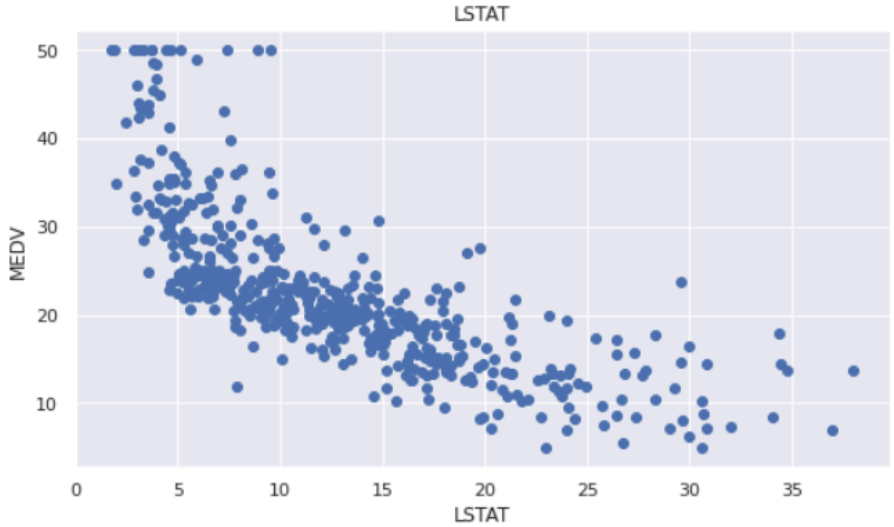
Empirical Risk Minimization

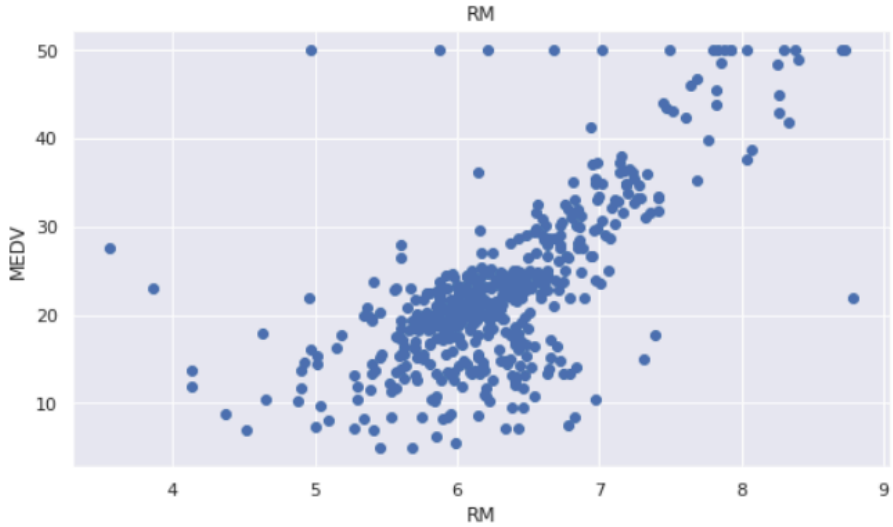
- So there are two things we need to determine:
  - Hypothesis class (function class) – The search space of function f, our model
  - The notation/measurement of error
- How? Judge from the data or the problem.
  - (no free lunch theory)

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Empirical Risk Minimization

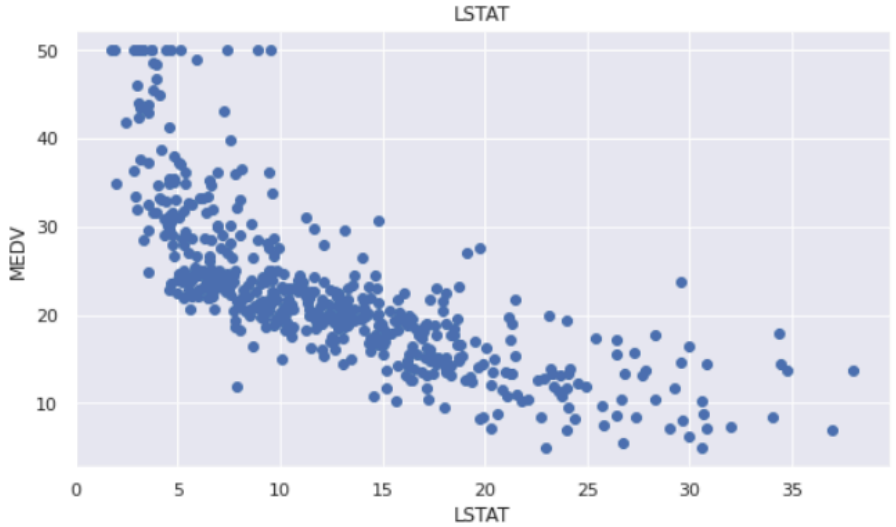
- So there are two things we need to determine:
  - Hypothesis class (function class) – The search space of function f, our model
  - The notation/measurement of error
- How? Judge from the data or the problem.
  - (no free lunch theory)

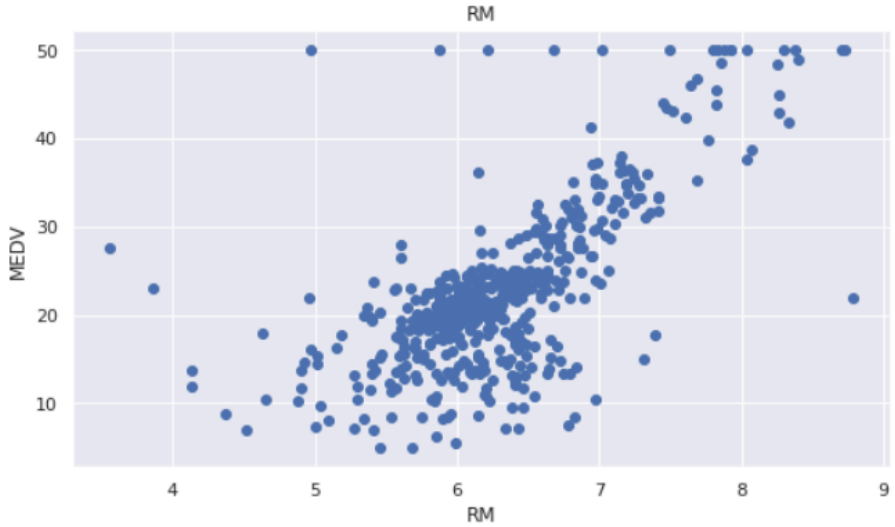




Empirical Risk Minimization

- So there are two things we need to determine:
  - Hypothesis class (function class) – The search space of function f, our model
  - The notation/measurement of error
- How? Judge from the data or the problem.
  - (no free lunch theory)
  - My personal preference: simplicity first





Linear Class/Hypothesis

- Suppose f is a linear function of x in the form of:

$$f_\theta(x) = w_0 + w_1 * x_1 + w_2 * x_2 + \ldots + w_m * x_m$$

$$= w_0 + \sum_{m}^{j} w_j * x_j$$

$$= w_0 + \boldsymbol{w}^T \boldsymbol{x}$$

- Where:
  - Parameters are $\theta = \{\boldsymbol{w}_0, \boldsymbol{w}\}$
    - The search space
  - w_0 is the bias term

Linear Class/Hypothesis

- Suppose f is a linear function of x in the form of:

$$f_\theta(x) = w_0 + w_1 * x_1 + w_2 * x_2 + \ldots + w_m * x_m$$

$$= w_0 + \sum_{m}^{j} w_j * x_j$$

$$= w_0 + \boldsymbol{w}^T \boldsymbol{x}$$

- Where:
  - Parameters are $\theta = \{\boldsymbol{w}_0, \boldsymbol{w}\}$
    - The search space
  - w_0 is the bias term

Empirical Risk Minimization

- So there are two things we need to determine:
  - Hypothesis class (function class) – The search space of function f, our model
  - The notation/measurement of error ?
  - The difference in the predicted pricing and the actual pricing

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Empirical Risk Minimization

- Suppose f is a linear function of x in the form of:

$$f_\theta(x) = w_0 + w_1 * x_1 + w_2 * x_2 + \ldots + w_m * x_m$$
$$= w_0 + \sum_{m}^{j} w_j * x_j$$
$$= w_0 + \boldsymbol{w}^T \boldsymbol{x}$$

- Then the least-square error is defined as:

$$Err(\theta) = \sum_{|D|}^{i=0} (y_i - f(\boldsymbol{x}_i))^2$$
$$= \sum_{|D|}^{i=0} (y_i - w_0 - \boldsymbol{w}^T \boldsymbol{x}_i)^2$$

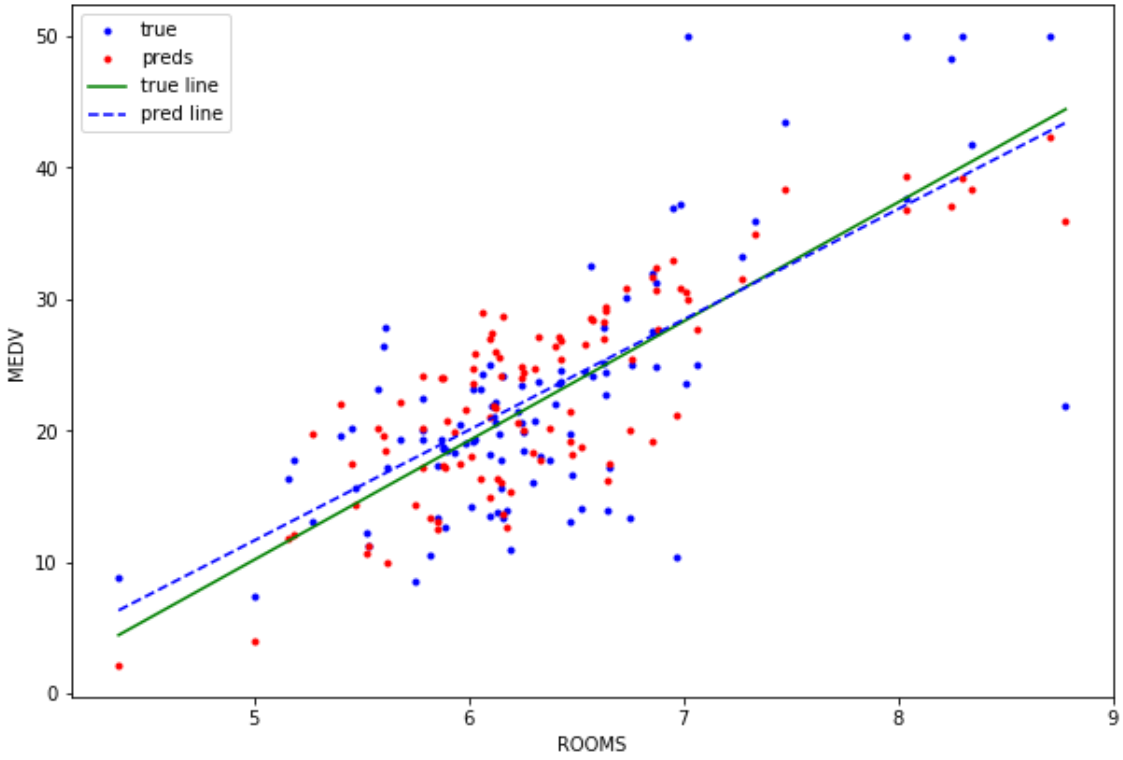
Empirical Risk Minimization

- Goal:
  - Find a function f such that
  - (in other words, find a parameterization of $\theta = \{w_0, \boldsymbol{w}\}$)

$$f_{best} = \arg\min Err(\theta)$$

$$= \sum_{|D|}^{i=0} (y_i - w_0 - \boldsymbol{w}^T \boldsymbol{x}_i)^2$$

Empirical Risk Minimization



$$f_{best} = \arg\min Err(\theta)$$

$$= \sum_{|D|}^{i=0} (y_i - w_0 - \boldsymbol{w}^T \boldsymbol{x}_i)^2$$

https://blog.goodaudience.com/linear-regression-on-the-boston-housing-data-set-d18c4ce4d0be

Solving with Linear Algebra

- YOU CAN SKIP

$$f_\theta = \boldsymbol{X}\boldsymbol{w}$$

$$Err(\theta) = (\boldsymbol{y} - \boldsymbol{w})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})$$

$$\partial Err(\theta)/\partial \boldsymbol{w} = -2\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})$$

$$\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) = 0$$

$$\boldsymbol{w}_{estimated} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$$

Solving with Linear Algebra

- So we have: $\boldsymbol{w}_{estimated} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$
  - AKA coefficient
  - How does increasing w[j] change the output y'
- We can also have the fitted y:

$$\boldsymbol{y}' = \boldsymbol{X}\boldsymbol{w}_{estimated}$$

- Based on the predicted y', we can calculate the standard error for each weight.
  - How confident/precise is the estimation of w[j] based on the training set?

Empirical Risk Minimization

Boston Housing Dataset

	coef	std err	t	P>|t|	[0.025	0.975]
CRIM	-0.0929	0.034	-2.699	0.007	-0.161	-0.025
ZN	0.0487	0.014	3.382	0.001	0.020	0.077
INDUS	-0.0041	0.064	-0.063	0.950	-0.131	0.123
CHAS	2.8540	0.904	3.157	0.002	1.078	4.630
NOX	-2.8684	3.359	-0.854	0.394	-9.468	3.731
RM	5.9281	0.309	19.178	0.000	5.321	6.535
AGE	-0.0073	0.014	-0.526	0.599	-0.034	0.020
DIS	-0.9685	0.196	-4.951	0.000	-1.353	-0.584
RAD	0.1712	0.067	2.564	0.011	0.040	0.302
TAX	-0.0094	0.004	-2.395	0.017	-0.017	-0.002
PTRATIO	-0.3922	0.110	-3.570	0.000	-0.608	-0.176
B	0.0149	0.003	5.528	0.000	0.010	0.020
LSTAT	-0.4163	0.051	-8.197	0.000	-0.516	-0.317

DATA MINING

/CISC 873 - Steven Ding
/Week #1/Lecture 2
Gradient Descent

Empirical Risk Minimization

- Suppose f is a linear function of x in the form of:

$$f_\theta(x) = w_0 + w_1 * x_1 + w_2 * x_2 + \ldots + w_m * x_m$$
$$= w_0 + \sum_{m}^{j} w_j * x_j$$
$$= w_0 + \boldsymbol{w}^T \boldsymbol{x}$$

- Then the least-square error is defined as:

$$Err(\theta) = \sum_{|D|}^{i=0} (y_i - f(\boldsymbol{x}_i))^2$$
$$= \sum_{|D|}^{i=0} (y_i - w_0 - \boldsymbol{w}^T \boldsymbol{x}_i)^2$$

Solving with Gradient Descent

- Linear Algebra Solution $\boldsymbol{w}_{estimated} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$
  - 3 matmul operation and 1 inversion
  - polynomial complexity on |D|
  - ~ O(|D|^3)
  - Not **scalable for large dataset** (also need all into memory at once)
- Gradient Decent:
  - Instead of directly getting the most accurate \theta directly from the dataset using matrix operations, gradually improve \theta to reduce the empirical error.
  - So we can have:

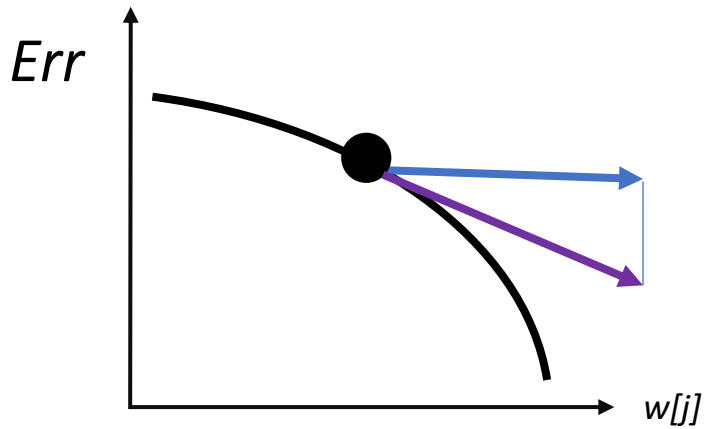
$$Err(w_0) \rightarrow Err(w_1) \rightarrow Err(w_2) \rightarrow Err(w_3)....$$

  - Ideally, the empirical errors are decreasing as we adjust \theta along the way. Here 0, 1, 2, 3, …, denotes the iterations.
  - We keep improving \theta until convergence

$$\left|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\right| \leq \epsilon$$

Solving with Gradient Descent

- Recall: $\partial Err(\theta)/\partial \boldsymbol{w} = -2\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})$
  - Given specific \theta we can know its gradients to improve it!
  - To improve = to reduce Err(…)
  - Complexity is ~ O(|D|^2)



- Algorithm:
  - Initialize \theta as w0
    - For t in range(n):

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \alpha \times \partial Err(\theta)/\partial \boldsymbol{w}_t$$

      - Stop if:

$$\left|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\right| \leq \epsilon$$

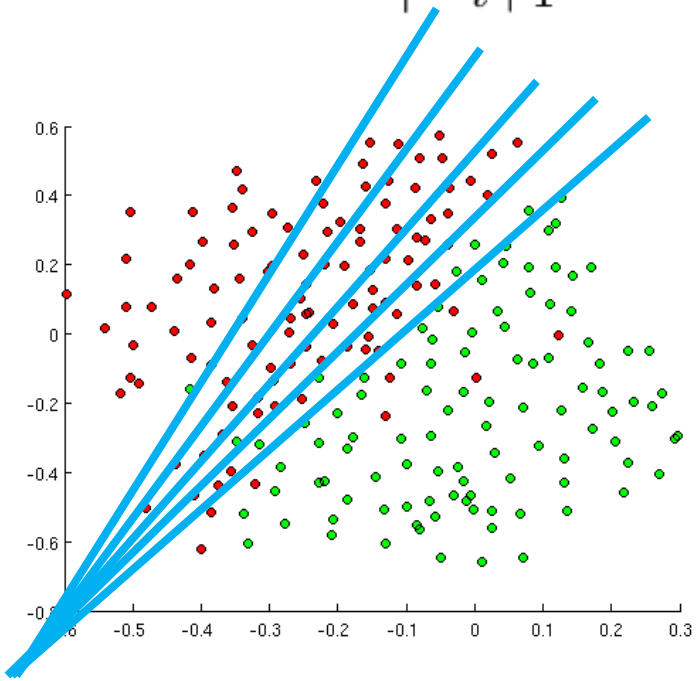
Solving with Gradient Descent

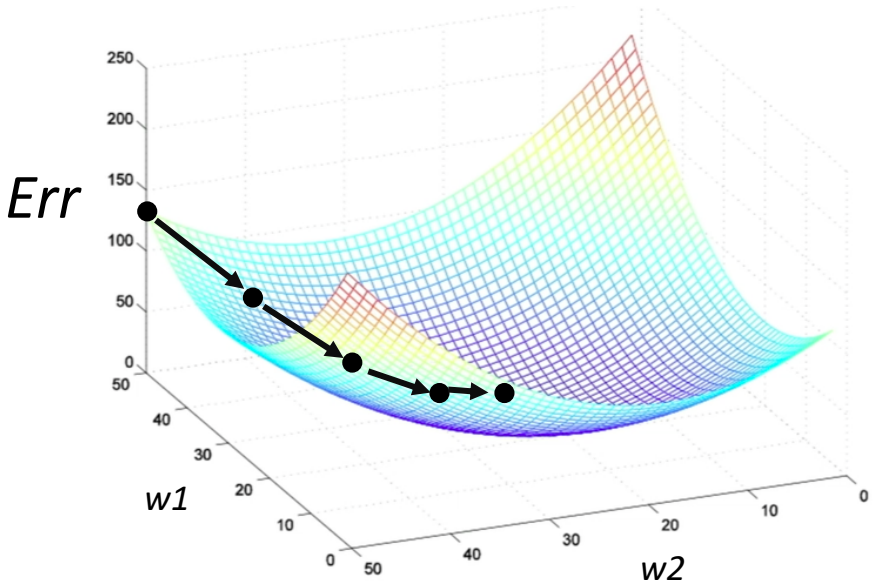
- Algorithm:
  - Initialize \theta as w0
    - For t in range(n):

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \alpha \times \partial Err(\theta)/\partial \boldsymbol{w}_t$$

    - Stop if:

$$\left|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\right| \leq \epsilon$$





Solving with Gradient Descent

- Algorithm:
  - Initialize \theta as w0
    - For t in range(n):

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \alpha \times \partial Err(\theta)/\partial \boldsymbol{w}_t$$

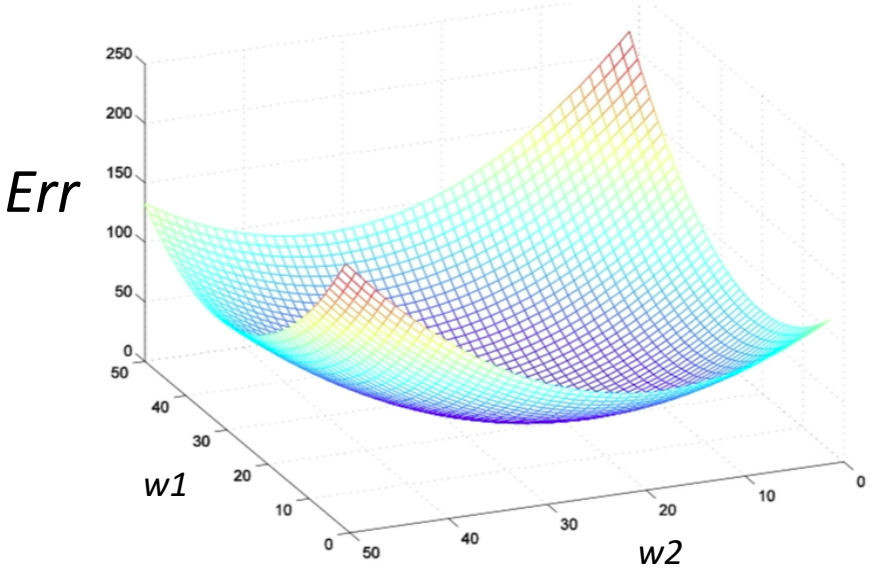
    - Stop if:

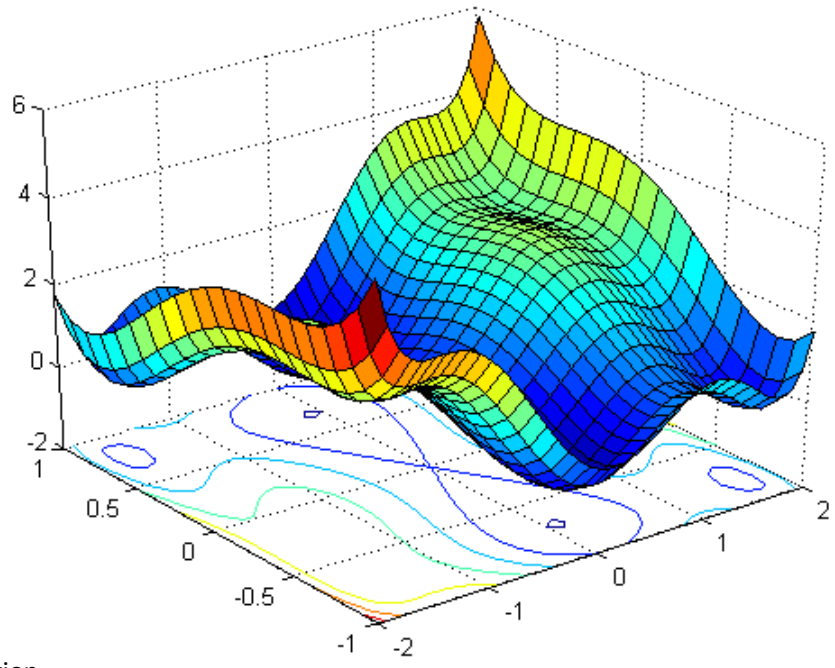
$$\left|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\right| \leq \epsilon$$

- Step size \alpha:
  - Control the `step` size to going down the direction

- Complexity:
  - O(|D|^2 * n). n << |D|

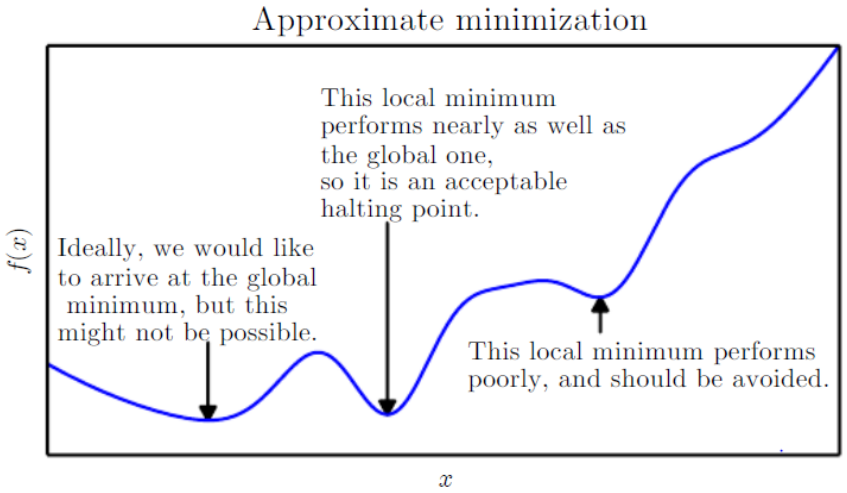
Convex vs Non-Convex Problem

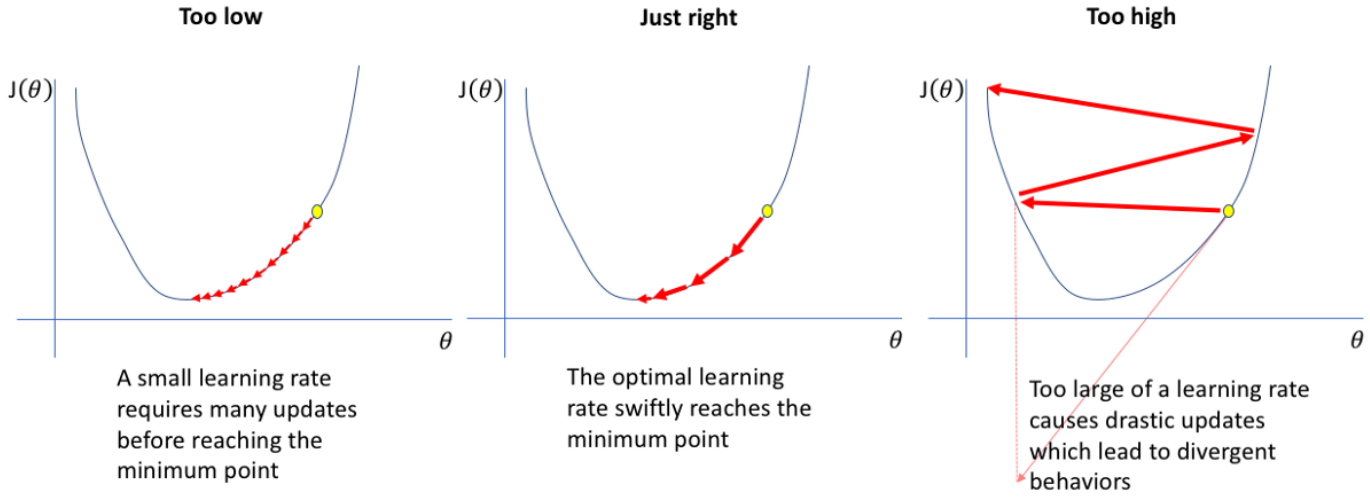




https://stats.stackexchange.com/questions/279292/non-convex-loss-function

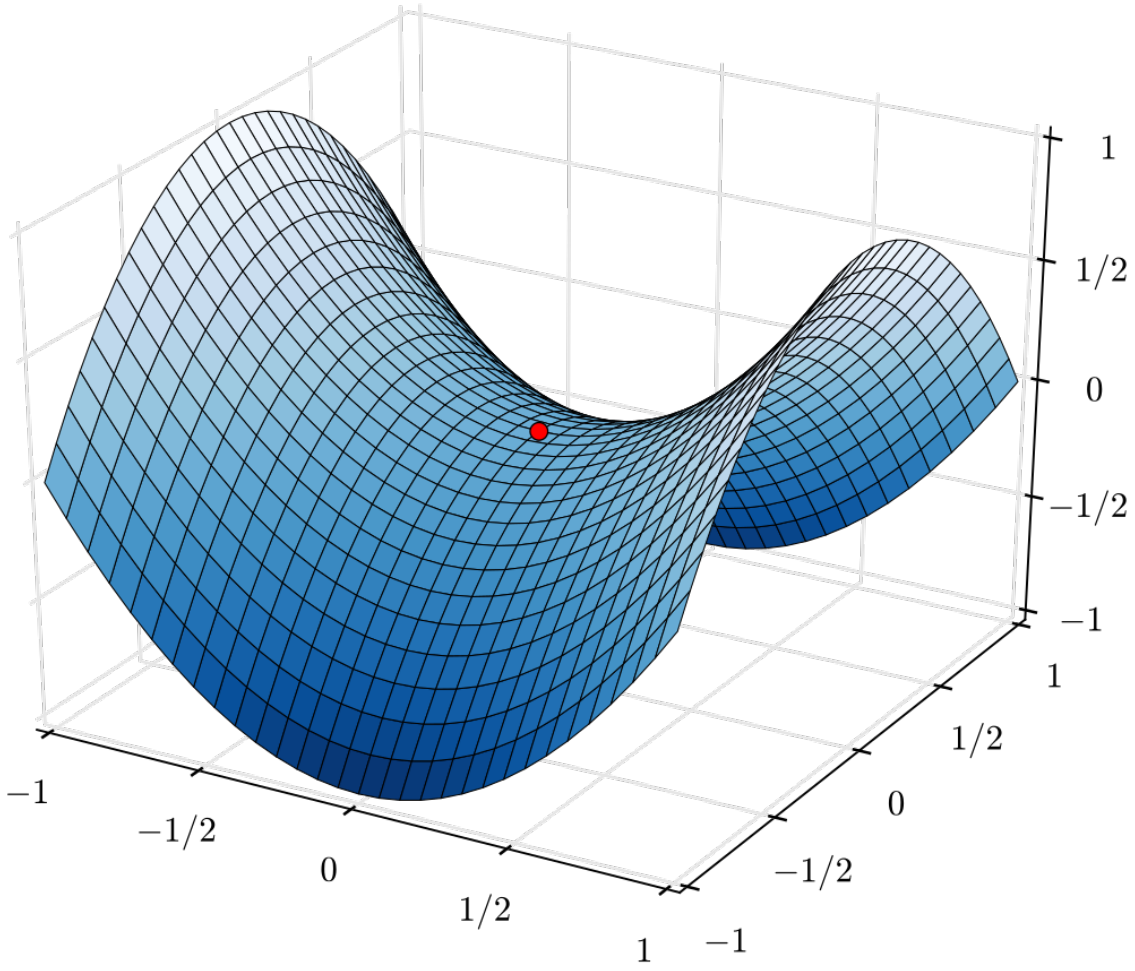
Global/Local Minima





https://medium.com/inveterate-learner/deep-learning-book-chapter-8-optimization-for-training-deep-models-part-i-20ae75984cb2

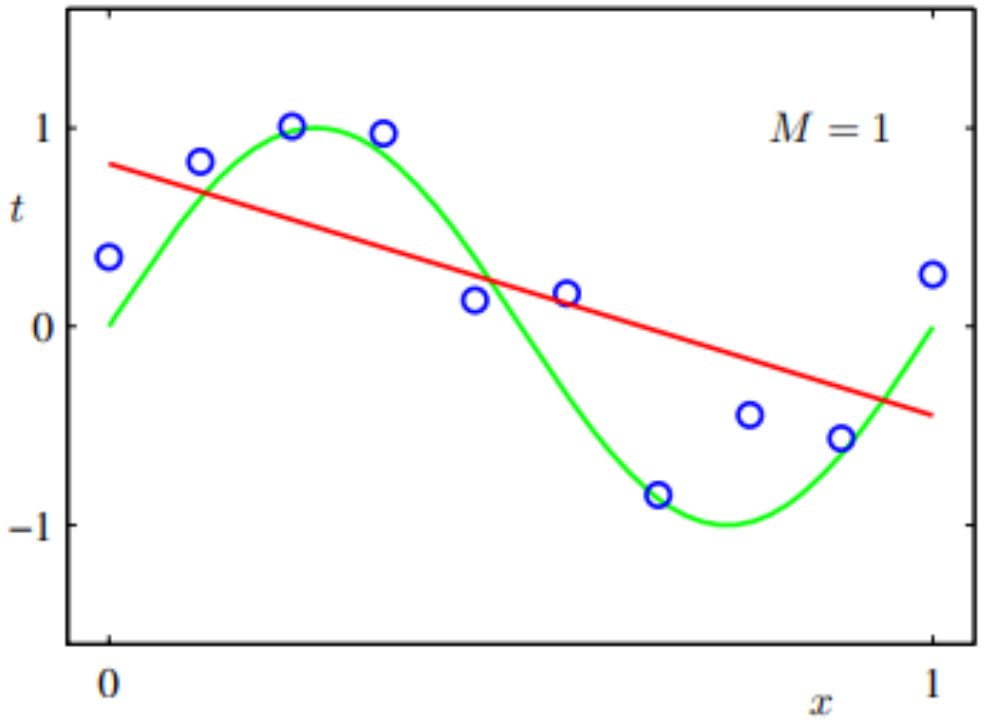
Saddle point



https://stats.stackexchange.com/questions/279292/non-convex-loss-function

Hypothesis class NOT good enough?

- Underfitting:
  - When current model/hypothesis cannot yield a good fit



Hypothesis class NOT good enough?

$x_i =< x_{i,1}, x_{i,2}, x_{i3}, ..., x_{i,m}, y_i >$

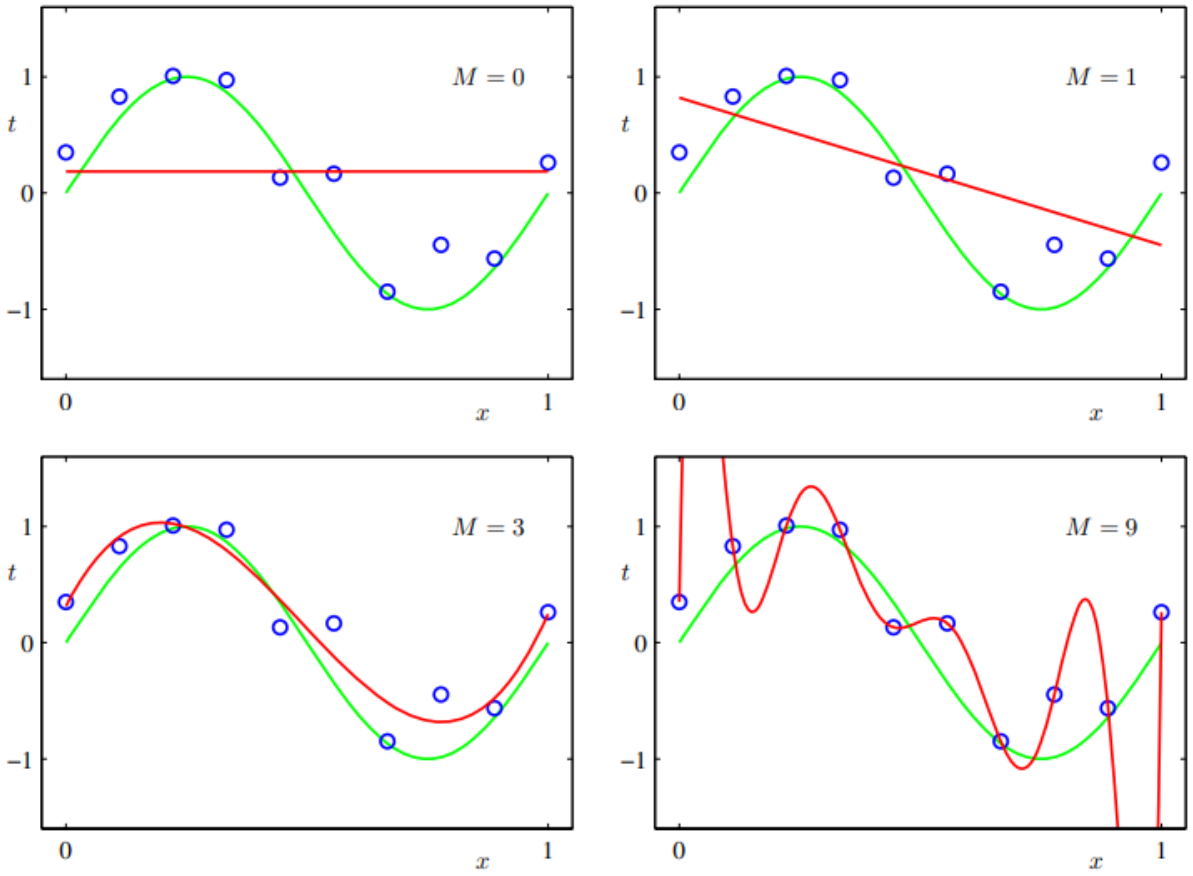
- $x_{i,j}$ is the value of the jth feature for the ith example.
- y_i is the value of the target attribute for the ith example.

- What can we do?

- Transform and add more features:
  - Basis functions: x^2, x^3
  - Transform the input space: log(x)
  - Interactions: x_i * x_j

Overfitting

- Introduce more degrees of freedom ==> always fit better (to the observations)



Bishop, C. M. (2006). Pattern recognition and machine learning. springer.

Overfitting

- Credits: Mahmoud Andelhadi

Regularization – (elastic net)

- L-1 Regularization (Lasso)
  - Panelize the absolute norm of parameters.
  - $\sum_{i=1}^{m} |w_k|$
  - Encourage model sparsity (turn on/off some features)

- L-2 regularization (Ridge)
  - Panelize the squares of parameters.
  - $\sum_{i=1}^{m} (w_k)^2$
  - Make the parameters small in scale.
  - Make the decision boundary less curved.

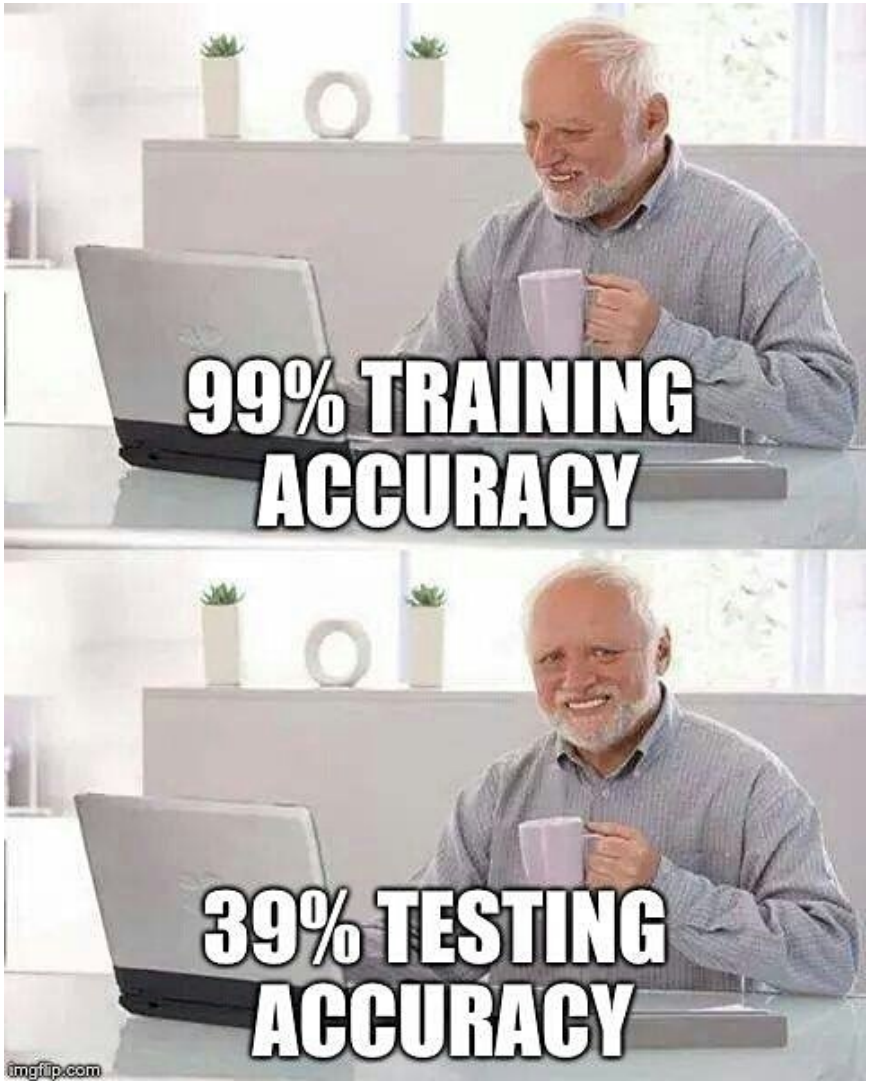
Generalizability

- How the trained model performs on unseen/future data
  - I.e. how generalizable is the trained model toward future unknown data (actual prediction tasks)

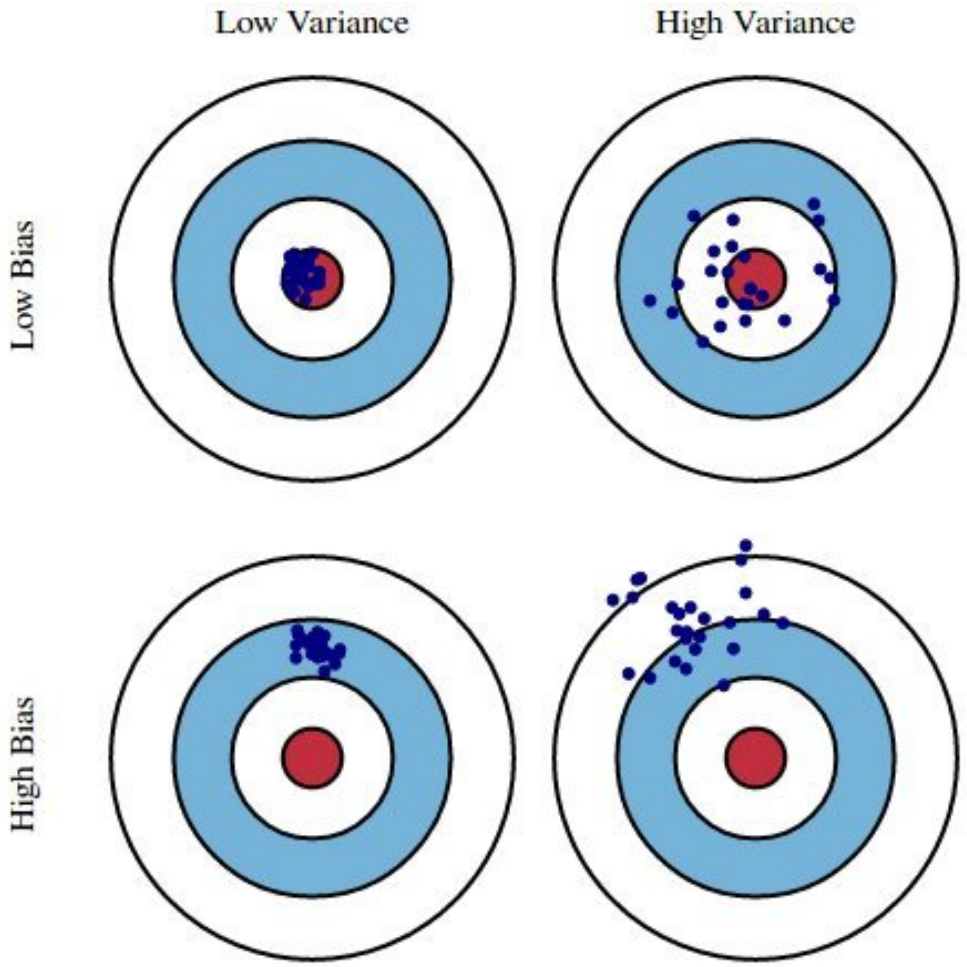
- Simulation:
  - Training set: for fitting the model
  - Testing set: for testing the model
  - Interpret the results on the testing set as the performance on unseen data

Overfitting

- Credits: ansariminhaj



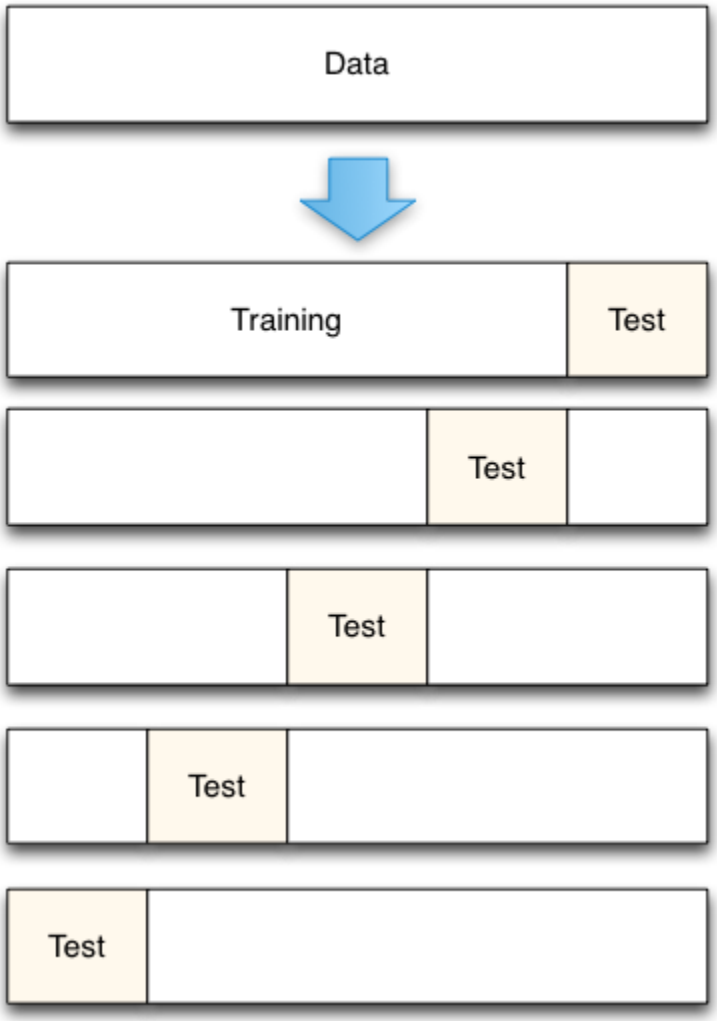
Bias vs Variance



bias is how removed a model's predictions are from correctness,

variance is the degree to which these predictions vary between model iterations.

K-fold cross-validation



Single test-train split (hold-out method): Estimation test error with **high variance**.

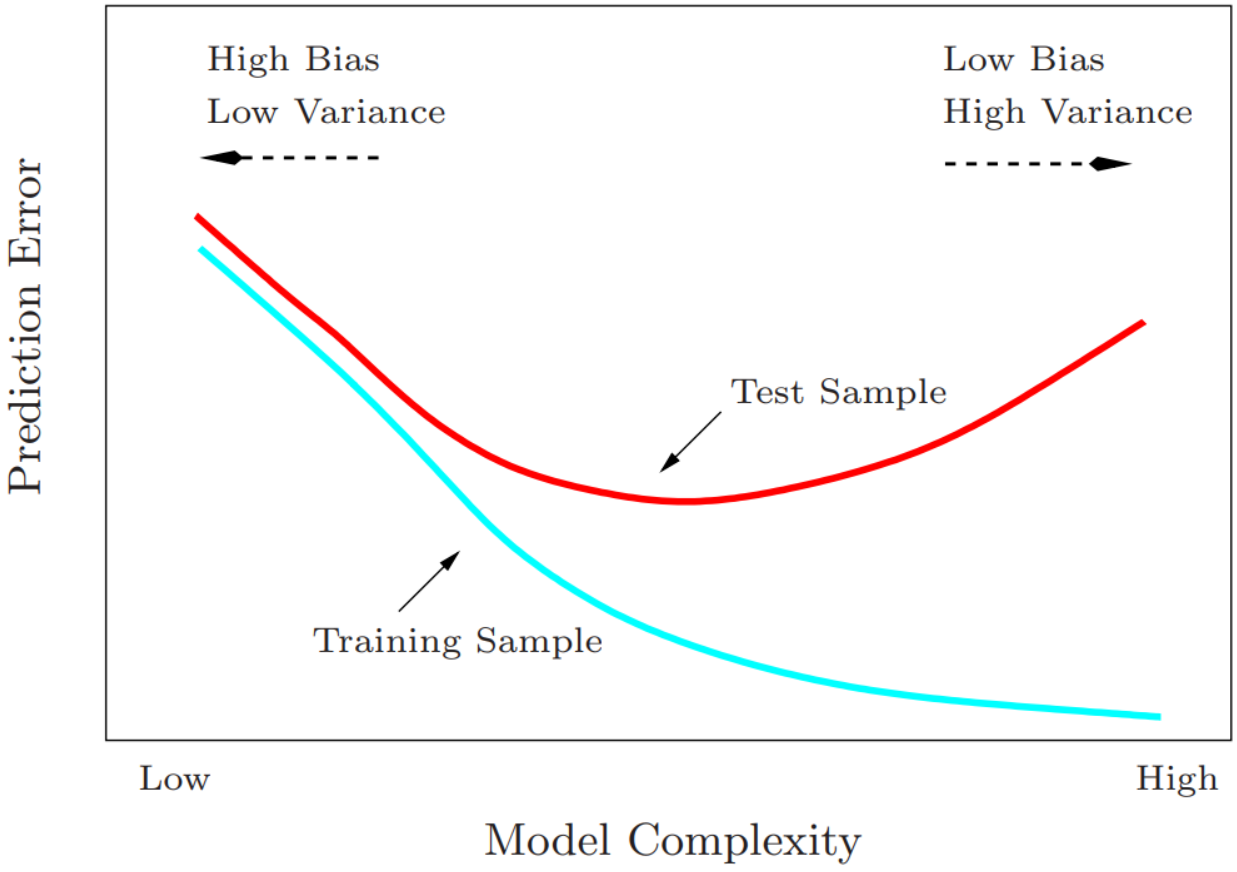
The lower the ***k***, the higher the bias in the error estimates and the less variance.

Conversely, when ***k*** is set equal to the number of instances, the error estimate is then very low in bias but has the possibility of high variance. (Leave-one-out cross-validation)

Cons: computational cost & waste of data

Understanding the Bias-Variance Tradeoff, by Scott Fortmann-Roe.

Biases & Variance Trade-off



Clarke, R., Coladarci, T., & Minium, E. W. (1999). Elements of statistical reasoning. Hoboken, NJ: John Wiley & Sons.

.632 Bootstrapping for error estimation

- Given a data set of ***d*** tuples. The data set is uniformly sampled ***d*** times, with replacement, resulting in a bootstrap sample set of ***d*** samples.
- The bootstrap sample set is used as training set.
- The data tuples that did not make it into the training set end up forming the test set.

.632 Bootstrapping – the resubstitution error

$$Acc(M) = \frac{1}{k}\sum_{i=1}^{k}(0.632 \times Acc(M_i)_{test\_set} + 0.368 \times Acc(M_i)_{train\_set}),$$

1. Each tuple has a probability of 1/d of being selected,
2. so the probability of not being chosen is (1 − 1/d).
3. We have to select d times, so the probability that a tuple will not be chosen during this whole time is (1 − 1/d) d .
4. If d is large, the probability approaches e −1 = 0.368.7
5. Thus, 36.8% of tuples will not be selected for training and thereby end up in the test set, and the remaining 63.2% will form the training set.

Efron, B., & Tibshirani, R. (1997). Improvements on cross-validation: the 632+ bootstrap method. *Journal of the American Statistical Association*, *92*(438), 548-560.

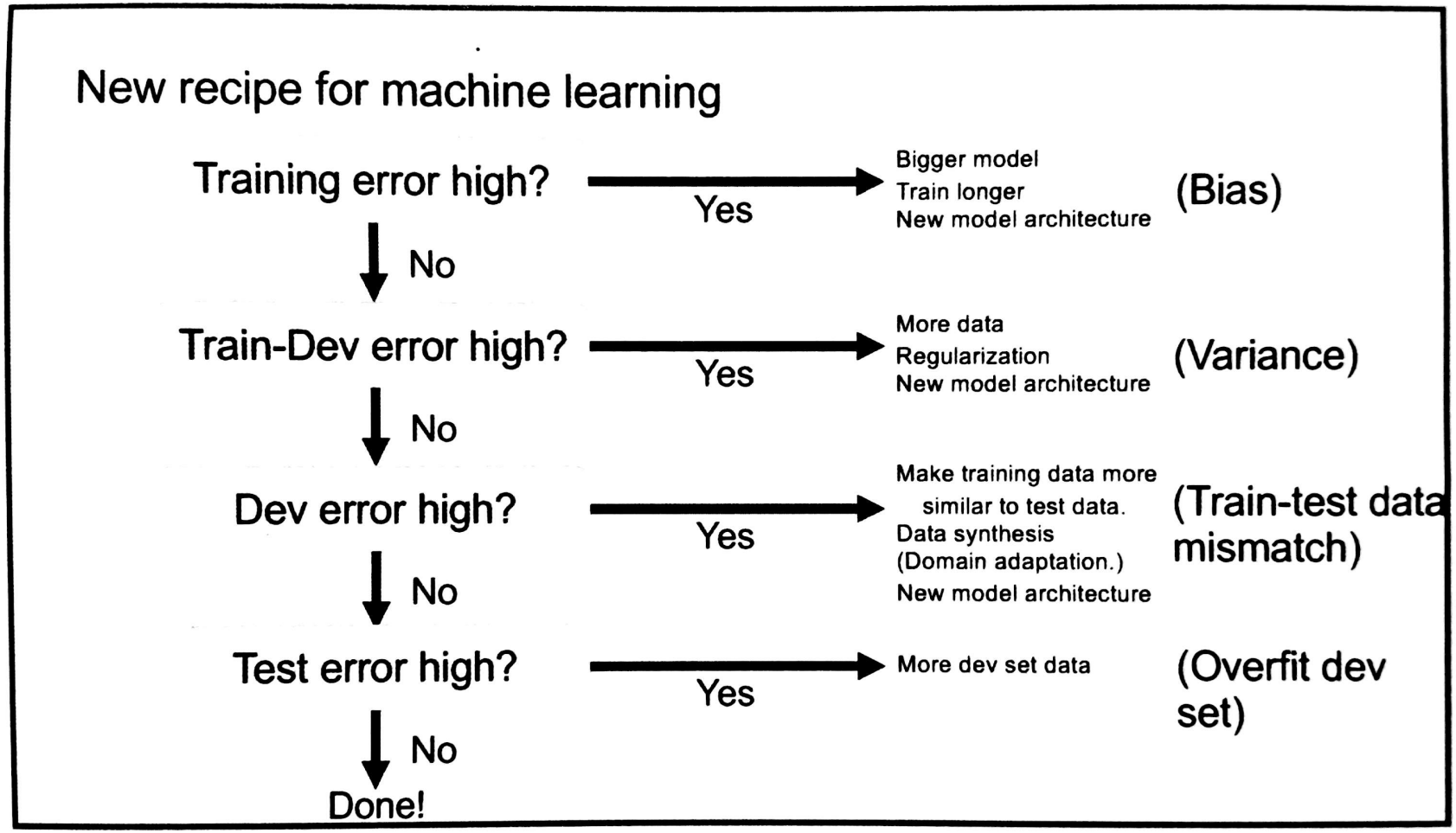
Hyper-parameter Optimization

- Classifier/Estimator has several configurable parameters
  - SVM: kernel type, C value, gamma value, etc.
  - Decision Tree: maximum depth, pruning conditions, etc.
  - Logistic Regression: regularization options, solver types, etc.
  - Selected based on our knowledge, data distribution, and empirical evaluation. (A process of educated guess and trial-and-error)
  - Tuning too hard on testing set may overfit the testing set.

Improved Hold-out Method:

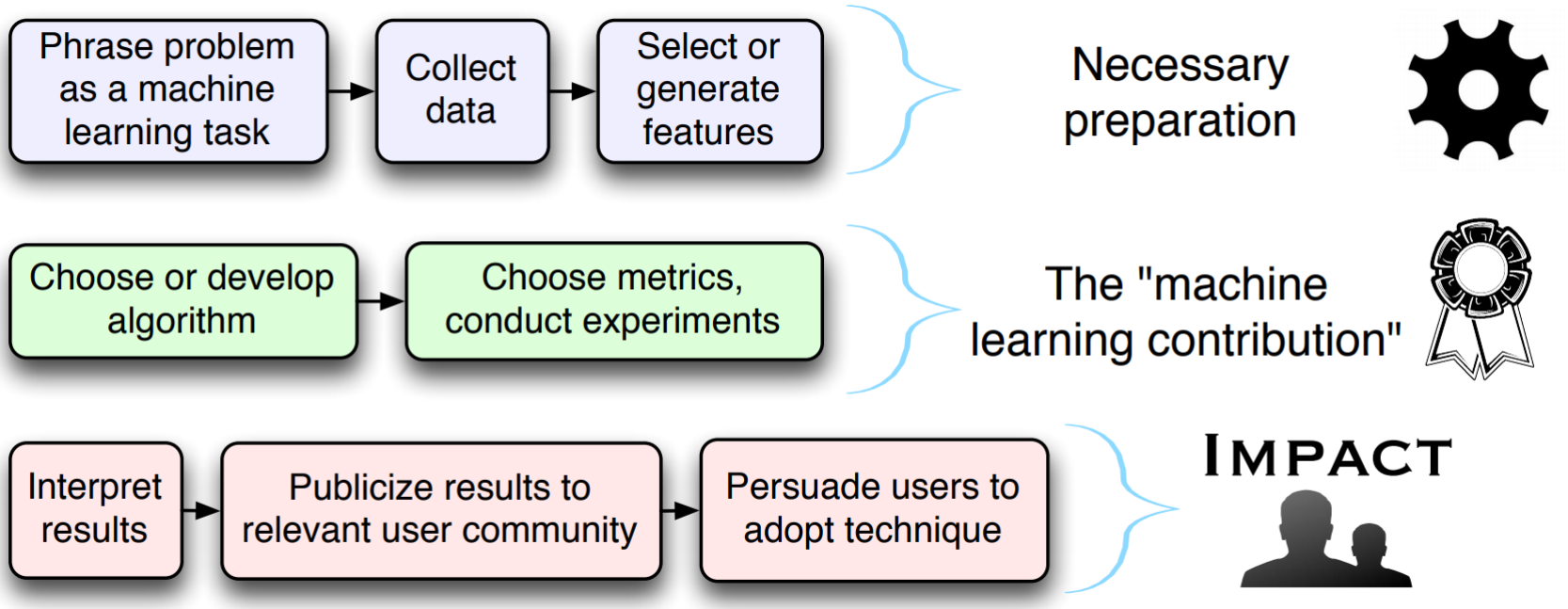
- Validation/Development Set
  - Partition the data into three independent sets: training set, validation set, and testing set.
    - Training set: used to train several classifiers based on different parameter configurations.
    - Validation set (1-2): used to evaluate the trained models. Pick the one that achieves the best performance.
    - Testing set: used to evaluate the chosen model and reports its performance. Minimal usage.

Data Science Life Cycle



Andrew Yan-Tak Ng

Data Science Project



https://www.wkiri.com/research/papers/wagstaff-MLmatters-12.pdf