Day 2: Linear Regression

Summer STEM: Machine Learning

Department of Electrical and Computer Engineering
NYU Tandon School of Engineering
Brooklyn, New York

Outline

Continuing on Vectorize Programming

Demo on vectorize programming.

Demo: Plotting Functions

- Generate and plot the following functions in Python:
  - Scatter plot of points: (0,1), (2,3), (5,2), (4,1)
  - Straight Line: $y = mx + b$
  - Sine-wave $y = sin(x)$
  - Polynomial e.g. $y = x^3 + 2$
  - Exponential e.g. $y = e^{-2x}$
  - Choose a function of your own
- Use Wikipedia and Numpy Documentation to search for mathematical formulas and python functions

Looking at our ice-breaker data in spreadsheets

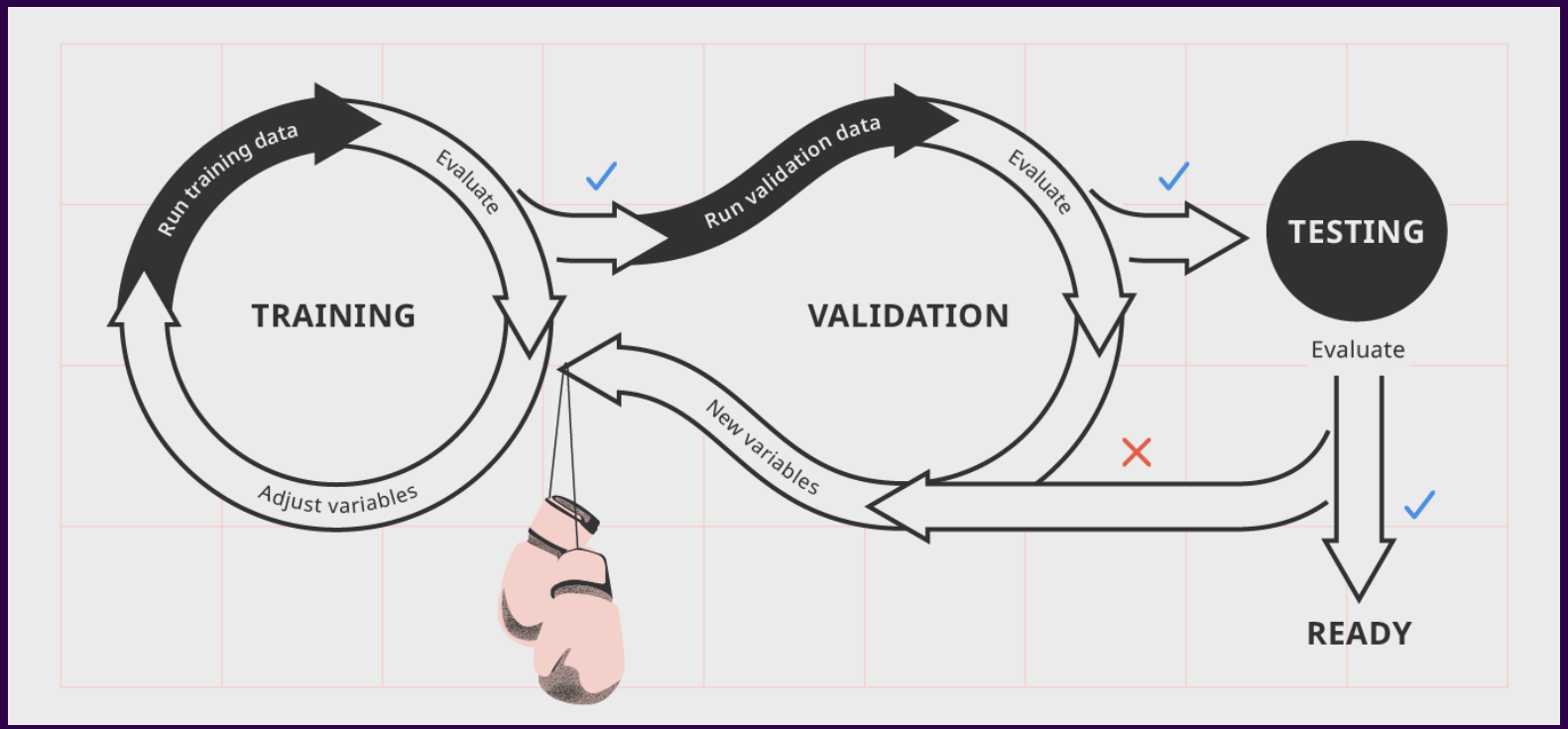
- Columns have labels in the first row
- Collected data (numbers, words) follow below
- Let's export it to a Comma-Separated Values (CSV) file and open it

Outline

What is Machine Learning

- Recognize patterns from data
- Make predictions based on the learnt patterns
- A very effective tool where human expertise is not available

Machine Learning Pipeline

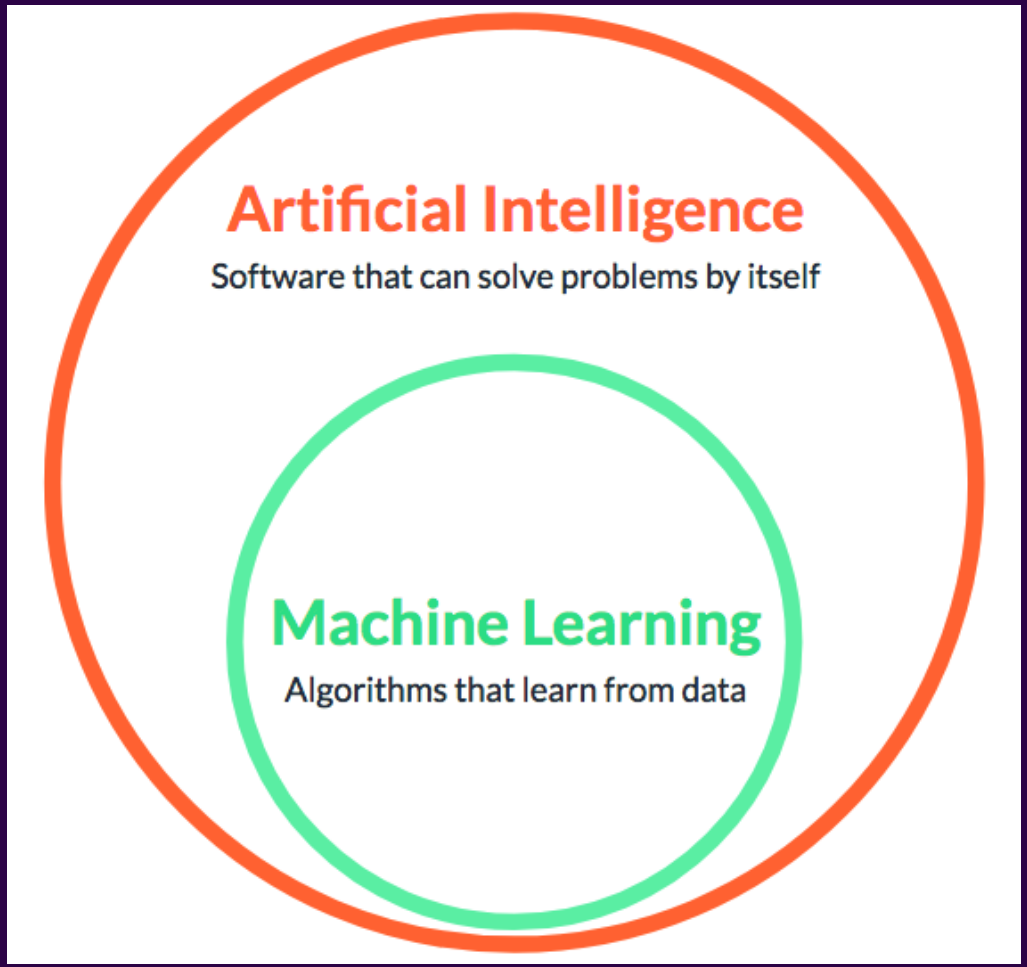


https://lionbridge.ai/articles/how-does-ai-training-work/

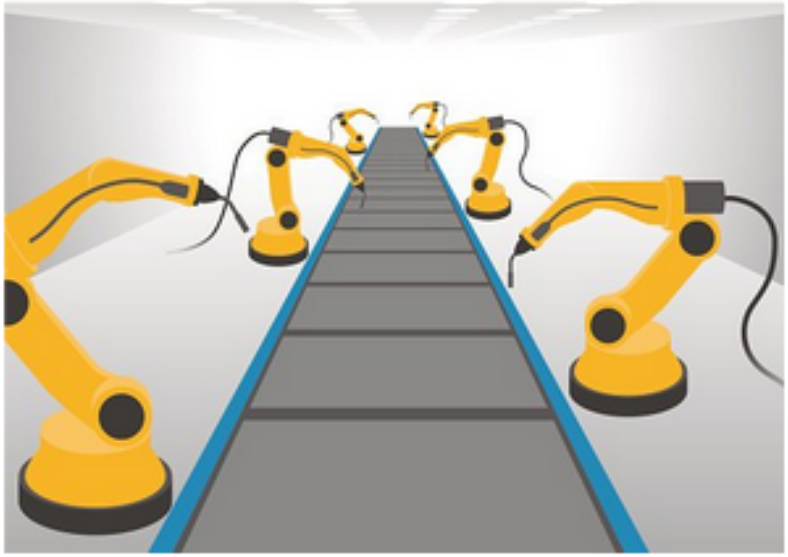
Artificial Intelligence

- Search
- Reasoning and Problem Solving
- Knowledge Representation
- Planning
- Learning
- Perception
- Natural Language Processing
- Motion and Manipulation
- Social and General Intelligence

Machine Learning



Autonomous vs. Automated



Autonomous Example: Self-driving car

- Waymo Video

https://www.tesmanian.com/blogs/tesmanian-blog/tesla-autopilot-full-self-driving-fsd-improvements-video

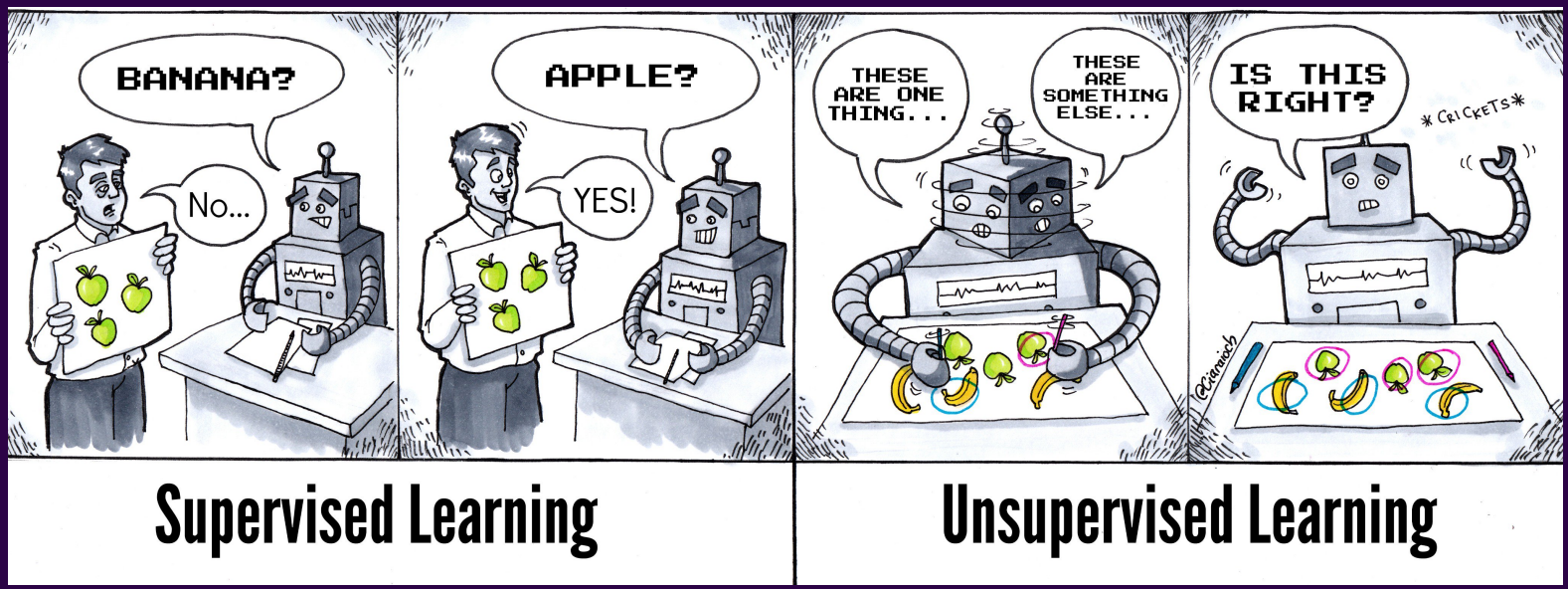
Why is Machine Learning so Prevalent?

- Database mining
- Medical records
- Computational biology
- Engineering
- Recommendation systems
- Understanding the human brain

Why Now?

- Big Data
  - Massive storage. Large data centers
  - Massive connectivity
  - Sources of data from internet and elsewhere
- Computational advances
  - Distributed machines, clusters
  - GPUs and hardware

Supervised Vs. Unsupervised Learning

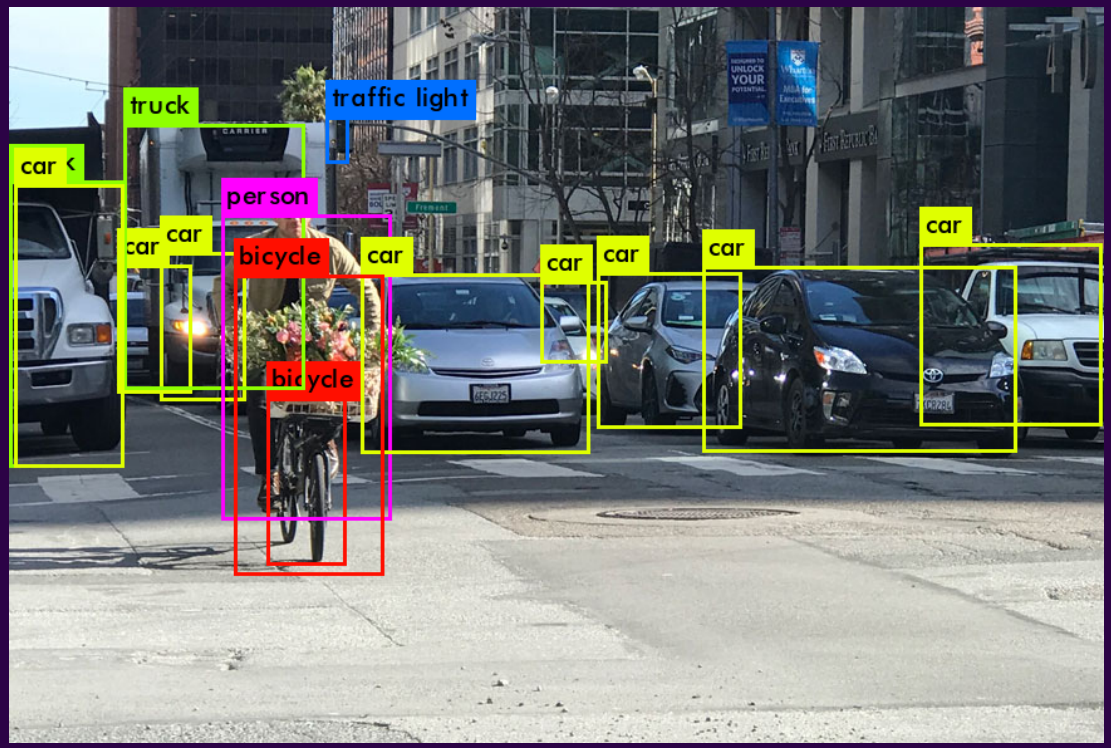


twitter.com/athena_schools/status/1063013435779223553/photo/1

Supervised Vs. Unsupervised Learning

- The main difference between supervised and unsupervised learning is the existence of a supervisor, which in many cases is in the form of a data label.
- The label of the data is what we want the machine learning algorithm to predict.

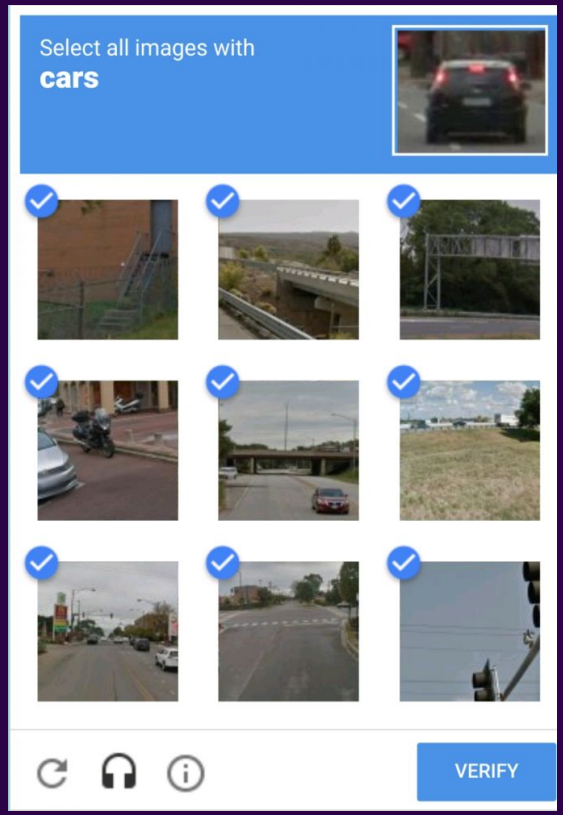
Labelled Data



- YOLO v2

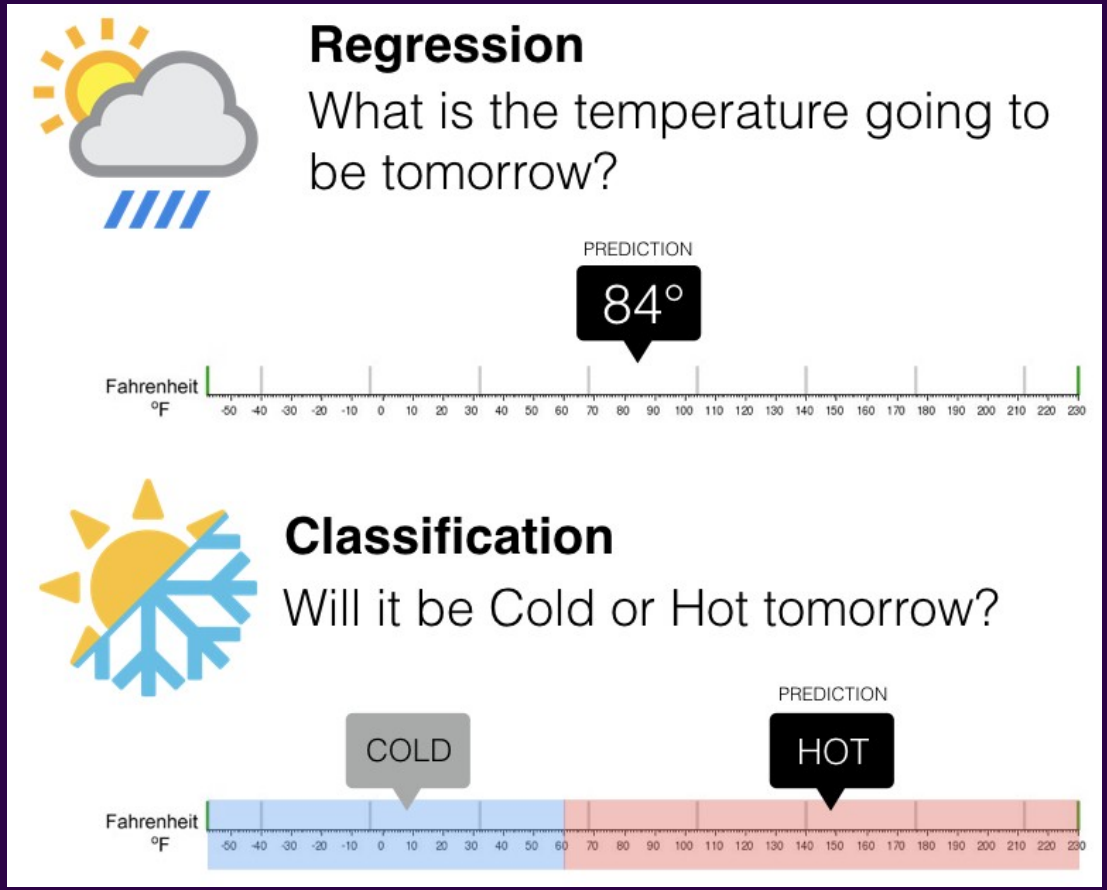
https://towardsdatascience.com/yolo-you-only-look-once-17f9280a4

How labels are generated



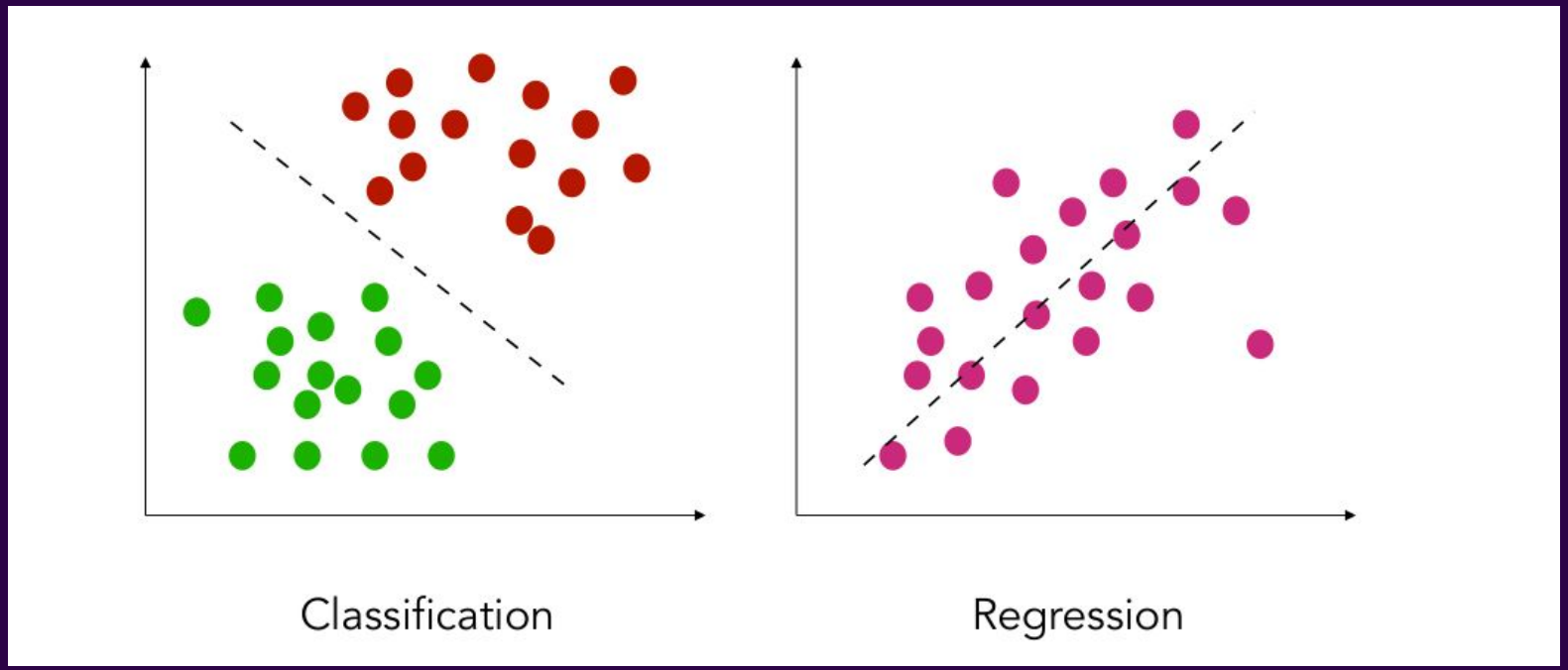
https://devrant.com/rants/1758134/select-all-images-with-cars-i-did-and-its-not-correct-why-not

Classification Vs. Regression



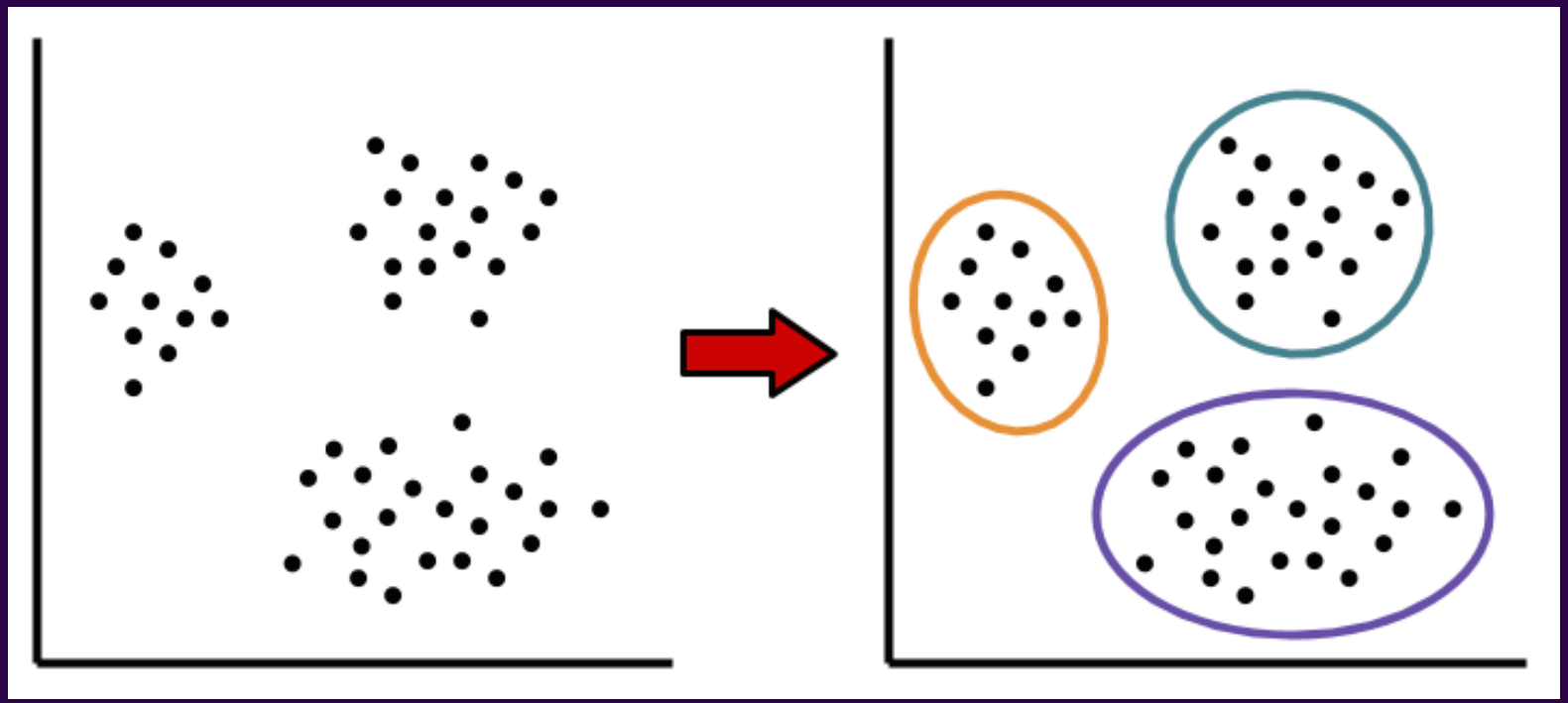
https://www.pinterest.com/pin/672232681855858622/?lp=true

Classification Vs. Regression



https://maelfabien.github.io/machinelearning/ml_base/

Unsupervised Learning



source: the dish on science

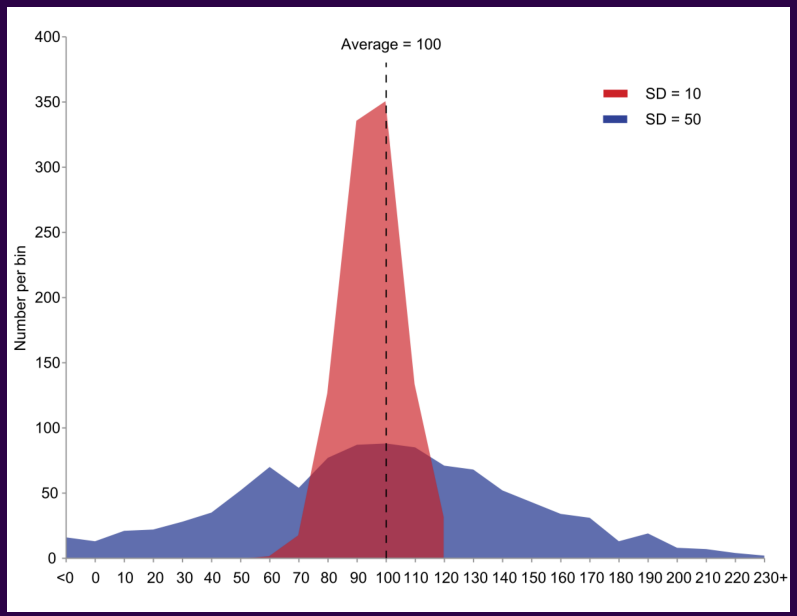
Outline

Basic Concepts

- **Mean** (average value): $\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i$
- **Variance** describes the spread of the data with respect to the mean.
- **Covariance** describes the relationship between two variables.

Variance

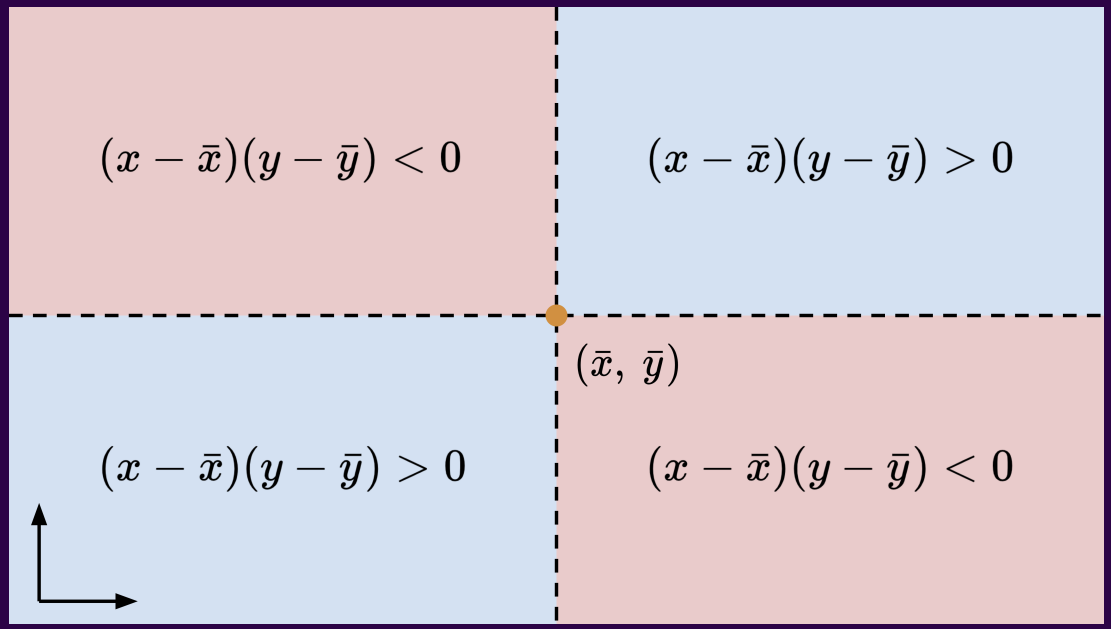
- Variance: $\sigma_x^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2$



https://en.wikipedia.org/wiki/Variance

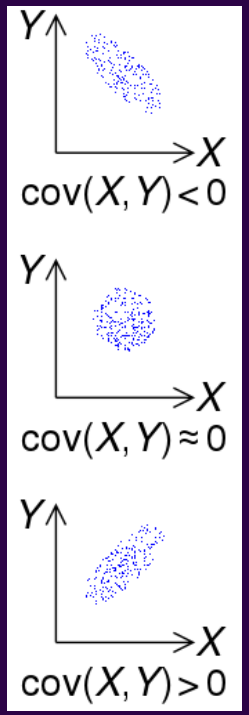
Covariance

- Covariance: $\sigma_{xy} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})$



Covariance

- Covariance: $\sigma_{xy} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})$



https://en.wikipedia.org/wiki/Covariance

Outline

Supervised learning in a nutshell

- I am sure that all of you are familiar with what $y = 2x + 1$ means.
- Here we introduce a new notation $f(x) = 2x + 1$.
- What this means is that we have a function $f(x)$ which has x as its variable.
- If we have different x values we will have different values of $f(x)$.

Supervised learning in a nutshell

- For $f(x) = 2x + 1$ and setting $x = 1$ we have $f(x) = 4$
- For $f(x) = 2x + 1$ and setting $x = 0$ we have $f(x) = 1$
- For $f(x) = 2x + 1$ and setting $x = -1.5$ we have $f(x) = -2$

Supervised learning in a nutshell

- We believe that dataset are representation of underlying models which can be represented as functions of features.
- For example, we can build a model to forecast weather, we can use the features humidity, current temperature and wind speed to estimate what the temperature will be tomorrow.
- Here we have $f(x)$ representing the tomorrow's temperature and x being a vector containing humidity, current temperature and wind speed.

Supervised learning in a nutshell

- But many times we do have $f(x)$ available, our task here is to figure out what f(x) is using the data available to us.
- Here $f(x)$ is called a model.
- In other words, we want to find a model that fits the data.

Supervised learning in a nutshell

- It would be easier to have a "framework" of the model ready and find the model parameters using the data.
- $f(x) = w_1 x + w_0$.
- $f(x) = w_2 x^2 + w_1 x + w_0$.
- $f(x) = \dfrac{1}{e^{-(w_1 x + w_0)} + 1}$.
- The numbers w_0, w_1 and w_2 are called model parameters.
- We often write the model as $f(x; \mathbf{w})$, stacking all parameters to a vector $\mathbf{w}$.

Structure of a dataset

- In a dataset we have many data.
- We can represent each piece of data as (x_i, y_i), $i = 1, 2, 3, \ldots$
- x_i is called the feature and y_i is called the label.
- The relationship between x_i and y_i and the model f is $f(x_i) \approx y_i$.
- Why "$\approx$" not "$=$"? Because the real world is not perfect like our models.
- For example, if the weather forecast says it will be $21^\circ C (69.8^\circ F)$ if it turns out to be $22^\circ C (71.6^\circ F)$ you won't be yelling at the TV.

How would you fit a line?

Can you find a line that passes through $(0, 0)$ and $(1, 1)$?

- The "framework" of the model is $f(x) = w_1 x + w_0$.
- The data is $(x = 0, f(x) = y = 0)$ and $(x = 1, f(x) = y = 1)$.
- The process of finding a model to fit the data is to find the values of w_1 and w_0.

How would you fit a quadratic curve?

Can you find a quadratic curve that passes through $(0, 0)$, $(1, 1)$ and $(-1, 1)$?

- The "framework" of the model is $f(x) = w_2x^2 + w_1x + w_0$.
- The data is $(x = 0, f(x) = y = 0)$, $(x = 1, f(x) = y = 1)$ and $(x = -1, f(x) = y = 1)$.
- The process of finding a model to fit the data is to find the values of w_2, w_1 and w_0.

What model do we use for this dataset?

- Open `demo_boston_housing_one_variable.ipynb`
- Can you find a line that go through ALL of the data points? Why?

Is Your Model a Good Fit?

- How would you determine if your model is a good fit or not?
  - How will you determine this?
  - Is there a quantitative way?
- We now introduce a new notation $f(x_i) = \hat{y}_i$ here the $\hat{\cdot}$ represents $f(x_i)$ is merely an estimation of y_i.

Error Functions

- An **error function** quantifies the discrepancy between your model and the data.
  - They are non-negative, and go to zero as the model gets better.
- Common Error Functions:
  - Mean Squared Error: $MSE = \frac{1}{N}\sum_{i=1}^{N} \|y_i - \hat{y}_i\|^2$
  - Mean Absolute Error: $MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i|$
- In later units, we will refer to these as **cost functions** or **loss functions**.
- Compute MSE on your model

Linear Regression

- Linear models: For scalar-valued feature x, this is $f(x) = w_1 x + w_0$
- One of the simplest machine learning model, yet very powerful.
- Two ways to get the solution, we will show them later.

Least Square Solution

- Model:

$$f(x) = w_0 + w_1 x$$

- Loss:

$$J(w_0, w_1) = \frac{1}{N} \sum_{i=1}^{N} \|y_i - f(x_i)\|^2$$

- Optimization: find w_0, w_1 such that $J(w_0, w_1)$ is the least possible value (hence the name "least square").

Least Square Solution: Using Statistics

- Solution:

$$f(x) = \bar{y} + \frac{\sigma_{xy}}{\sigma_x^2}(x - \bar{x})$$

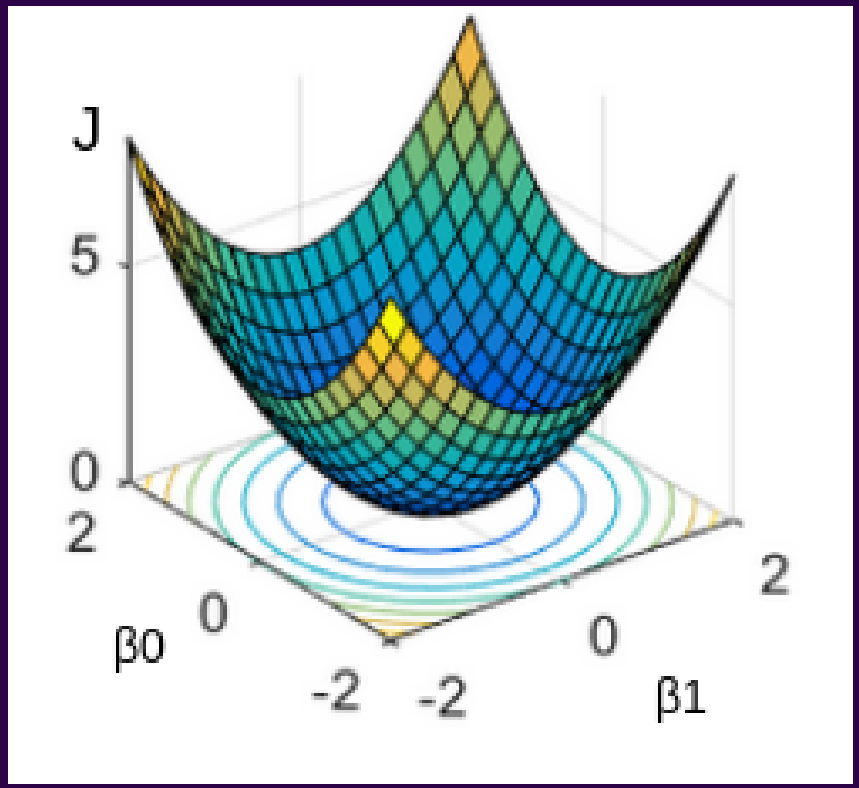
$$w_1 = \frac{\sigma_{xy}}{\sigma_x^2}, \quad w_0 = \bar{y} - w_1\bar{x}$$

- Prediction:

$$f(x) = w_0 + w_1 x$$

Loss Landscape

Plot the loss against the parameters:



Least Square Solution: Using Pseudo-Inverse

- For N data points (x_i, y_i) we have,

$$
\begin{aligned}
y_1 &\approx w_0 + w_1 x_1 \\
y_2 &\approx w_0 + w_1 x_2 \\
&\vdots \\
y_N &\approx w_0 + w_1 x_N .
\end{aligned}
$$

Linear Regression

- In matrix form we have,

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \approx \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}$$

- We can write it as $Y \approx X\mathbf{w}$. We call X the design matrix.
- Exercise: verify $\|Y - X\mathbf{w}\|^2 = \sum_{i=1}^{N} \|y_i - (w_0 + w_1 x_i)\|^2$

Linear Least Square

- $\min_{\mathbf{w}} \frac{1}{N} \|Y - X\mathbf{w}\|^2$
- Using the pseudo-inverse (only square matrices have an inverse),

$$
\begin{aligned}
Y &\approx X\mathbf{w} \\
X^T Y &\approx X^T X\mathbf{w} \\
(X^T X)^{-1} X^T Y &\approx (X^T X)^{-1} X^T X\mathbf{w} \\
(X^T X)^{-1} X^T Y &\approx \mathbf{w}.
\end{aligned}
$$

Linear Regression

- What if we have multivariate data with $\mathbf{x}$ being a vector?
- Ex: $\mathbf{x}_i = [x_{i1}, x_{i2}]^T$

$$
\begin{aligned}
y_1 &\approx w_0 + w_1 x_{11} + w_2 x_{12} = \hat{y}_1 \\
y_2 &\approx w_0 + w_1 x_{21} + w_2 x_{22} = \hat{y}_2 \\
&\vdots \\
y_N &\approx w_0 + w_1 x_{N1} + w_2 x_{N2} = \hat{y}_N
\end{aligned}
$$

- The model can be written as $\hat{y}_i = \mathbf{w}^T \phi(\mathbf{x}_i)$, here both $\mathbf{w} = [w_0, w_1, w_2]^T$ and $\mathbf{x}$ are vectors. $\phi(\mathbf{x})$ is a feature transformation that transforms the original feature to $\phi(\mathbf{x}_i) = [1, x_{i1}, x_{i2}]^T$.

Multilinear Regression

- In matrix-vector form

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \approx \begin{bmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_{n2} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix}$$

- Solution remains the same $(X^T X)^{-1} X^T Y = \mathbf{w}$
- Exercise: open `demo_multilinear.ipynb`