



---

Launching into ML



# Agenda

---

## **Python notebooks in the Cloud**

Supervised Learning

Inclusive ML

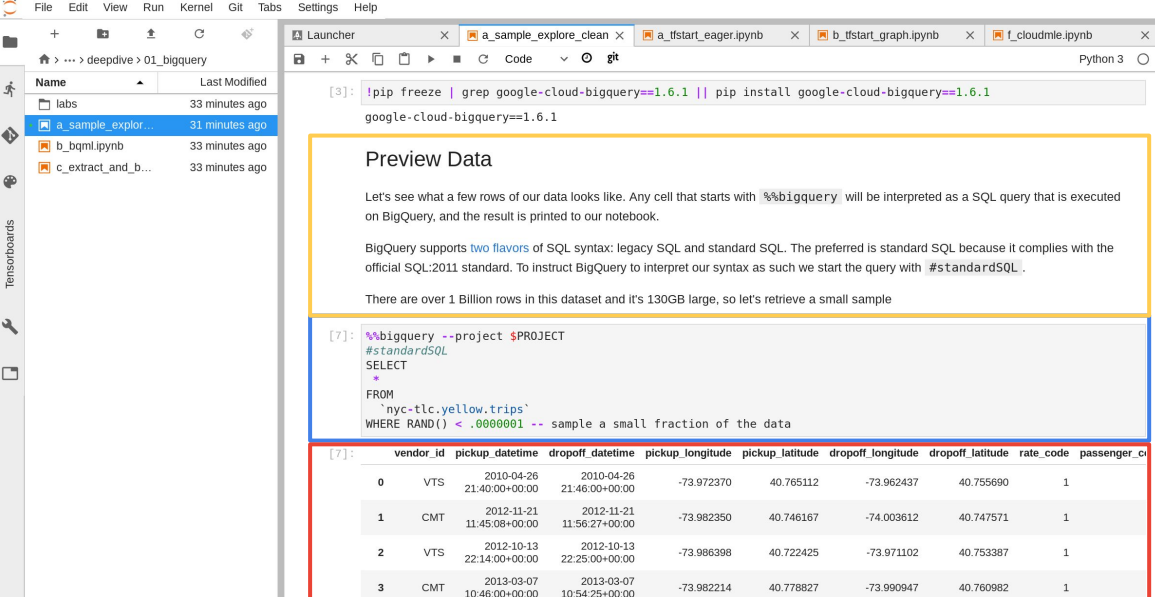
Short History of ML

# Increasingly, data analysis and ML are carried out in self-descriptive, shareable, executable notebooks

Markup

Code

Output



The screenshot shows a Jupyter Notebook interface. On the left is a file explorer showing a directory structure with files like 'a\_sample\_explor...', 'b\_bqml.ipynb', and 'c\_extract\_and\_b...'. The main area is divided into three sections: a code editor at the top, a 'Preview Data' section in the middle, and a table of results at the bottom. The code editor contains a pip freeze command and a BigQuery query. The 'Preview Data' section explains that the query is executed on BigQuery and shows a sample of the results. The table at the bottom displays the results of the query, with columns for vendor\_id, pickup\_datetime, dropoff\_datetime, pickup\_longitude, pickup\_latitude, dropoff\_longitude, dropoff\_latitude, rate\_code, and passenger\_count.

```
[3]: !pip freeze | grep google-cloud-bigquery==1.6.1 || pip install google-cloud-bigquery==1.6.1
google-cloud-bigquery==1.6.1
```

### Preview Data

Let's see what a few rows of our data looks like. Any cell that starts with `%bigquery` will be interpreted as a SQL query that is executed on BigQuery, and the result is printed to our notebook.

BigQuery supports *two flavors* of SQL syntax: legacy SQL and standard SQL. The preferred is standard SQL because it complies with the official SQL:2011 standard. To instruct BigQuery to interpret our syntax as such we start the query with `#standardSQL`.

There are over 1 Billion rows in this dataset and it's 130GB large, so let's retrieve a small sample

```
[7]: %bigquery --project $PROJECT
#standardSQL
SELECT
  *
FROM
  `nyc-tlc.yellow.trips`
WHERE RAND() < .000001 -- sample a small fraction of the data
```

	vendor_id	pickup_datetime	dropoff_datetime	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude	rate_code	passenger_count
0	VTS	2010-04-26 21:40:00+00:00	2010-04-26 21:46:00+00:00	-73.972370	40.765112	-73.962437	40.755690	1	
1	CMT	2012-11-21 11:45:08+00:00	2012-11-21 11:56:27+00:00	-73.982350	40.746167	-74.003612	40.747571	1	
2	VTS	2012-10-13 22:14:00+00:00	2012-10-13 22:25:00+00:00	-73.986398	40.722425	-73.971102	40.753387	1	
3	CMT	2013-03-07 10:46:00+00:00	2013-03-07 10:54:25+00:00	-73.982214	40.778827	-73.990947	40.760982	1	

A typical notebook contains code, charts, and explanations.



# Follow-along: The Easy Way to Make a Notebook

CLME notebooks are found under [ML Engine -> Notebook Instances](#)

Click “+ New Instances” then TensorFlow -> Standard

Install NVIDIA GPU drivers automatically. The Click “OPEN JUPYTER LAB” after the VM is spun up.

Google Cloud Platform My First Project

ML Engine Notebook instances BETA

Create and use Jupyter Notebooks with a JupyterLab pre-installed and are configured frameworks. [Learn more](#)

Filter table

No notebook instances to display

+ NEW INSTANCE REFRESH START STOP RESET DELETE

TensorFlow ▶ Standard us-west1-b, 4vCPUs, 15GB Memory, 100GB disk

PyTorch ▶ With GPU us-west1-b, 4vCPUs, 15GB Memory, 1 NVIDIA Tesla K80, 100GB disk

More options

Labels help organize your resources env:prod

Empty Tab

Notebook instances BETA + NEW INSTANCE REFRESH START

Filter table

Instance name	Region	ML framework	Machine type	Lab
tensorflow-20190307-214633	us-west1-b	TensorFlow	4 vCPU, 15GB RAM	

OPEN JUPYTER LAB

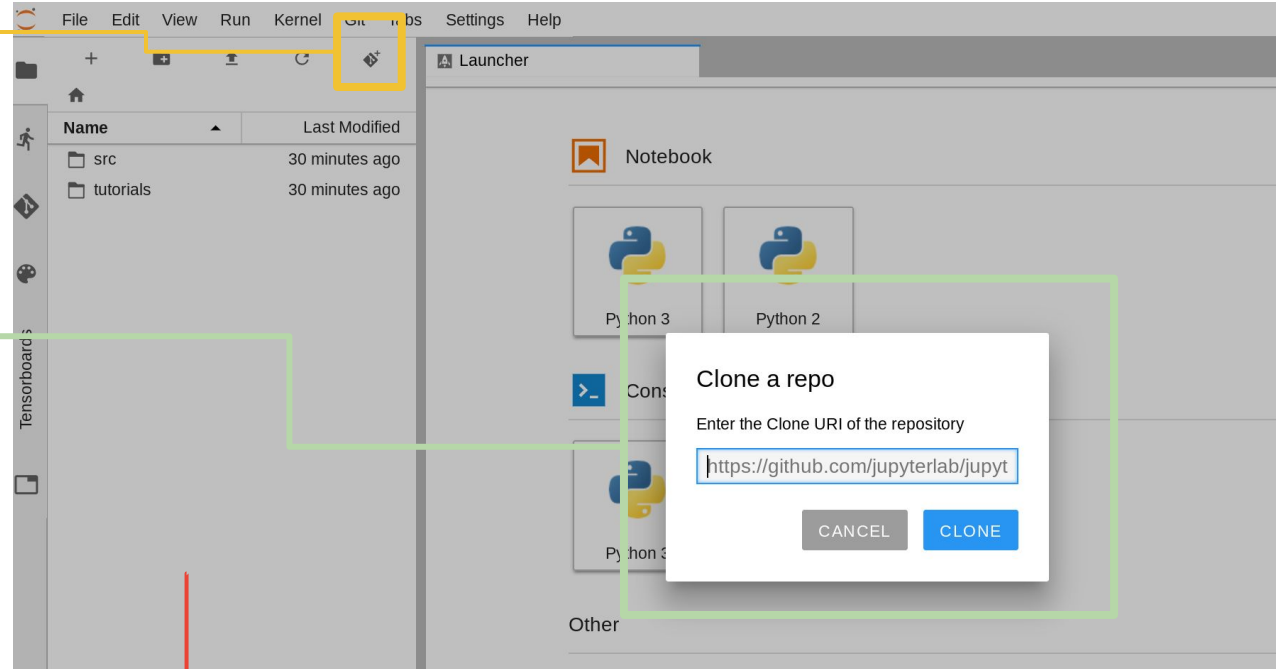
# Follow-along: Connecting to Github

Click the git clone icon to clone a repository

Paste the following URL into the address box and click "Clone"

<https://github.com/GoogleCloudPlatform/training-data-analyst.git>

Double click the "training-data-analyst" folder when it appears here.



# Agenda

---

Python notebooks in the Cloud

**Supervised Learning**

Inclusive ML

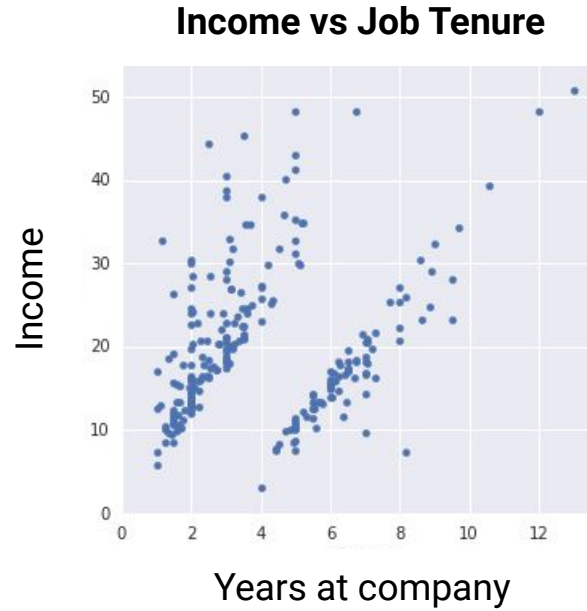
Short History of ML

# Unsupervised and supervised learning are the two types of ML algorithms

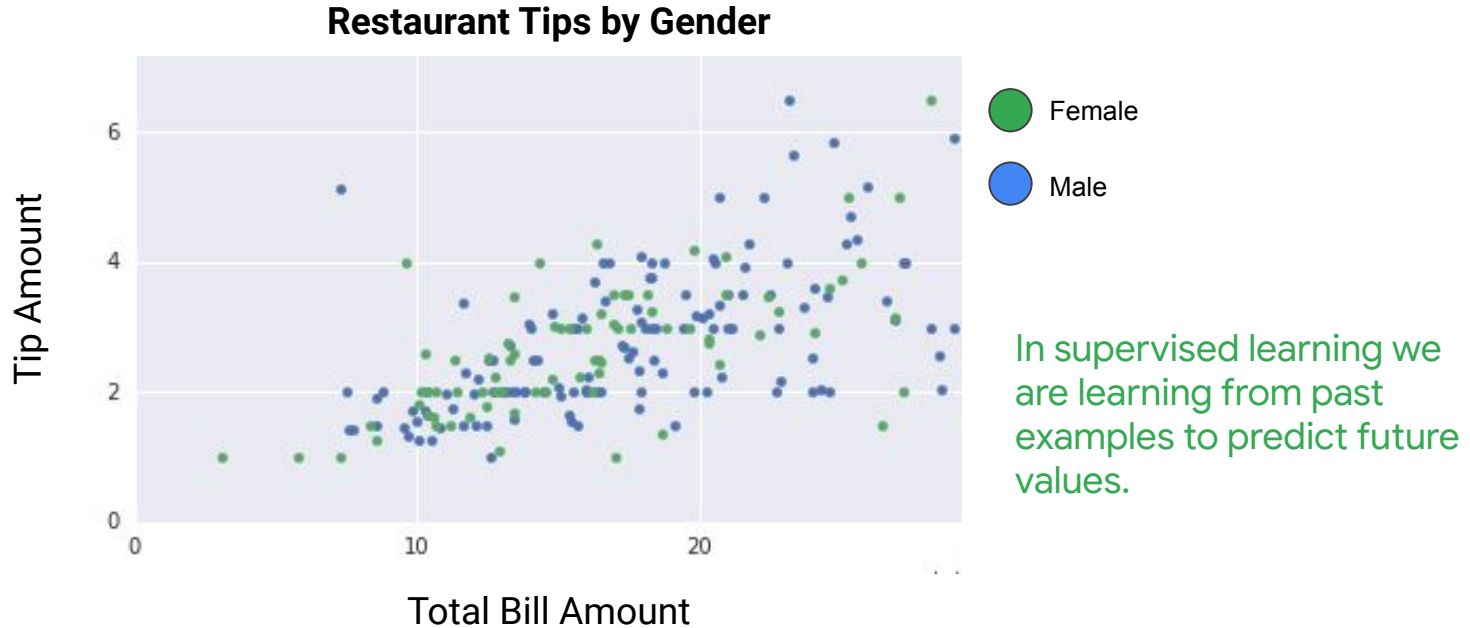
## Example Model: Clustering

Is this employee on the  
“fast-track” or not?

In unsupervised  
learning, data is not  
labeled.



# Supervised learning implies the data is already labeled





# Regression and classification are supervised ML model types

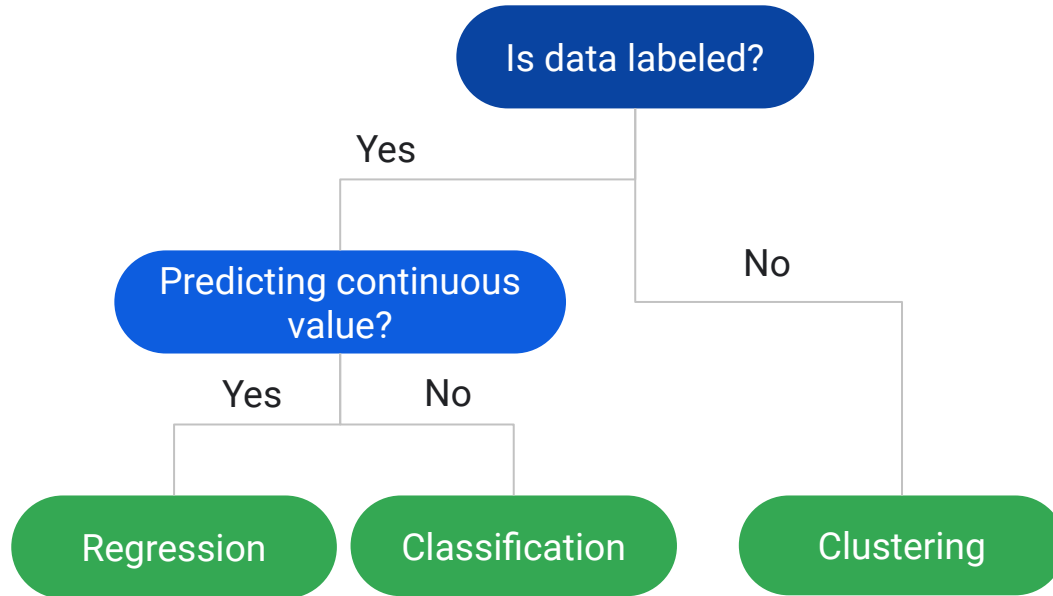
1	total_bill	tip	sex	smoker	day	time
2	16.99	1.01	Female	No	Sun	Dinner
3	10.34	1.66	Male	No	Sun	Dinner
4	21.01	3.5	Male	No	Sun	Dinner
5	23.68	3.31	Male	No	Sun	Dinner
6	24.59	3.61	Female	No	Sun	Dinner
7	25.29	4.71	Male	No	Sun	Dinner
8	8.77	2	Male	No	Sun	Dinner
9	26.88	3.12	Male	No	Sun	Dinner

**Option 1**  
**Regression Model**  
Predict the tip amount

**Option 2**  
**Classification Model**  
Predict the sex of the customer



The type of ML problem depends on whether or not you have labeled data and what you are interested in predicting



# Quiz: Supervised learning

Imagine you are in banking and you are creating an ML model for detecting if transactions are fraudulent or not. Is this classification or regression and why?

- A. Regression, categorical label
- B. Regression, continuous label
- C. Classification, categorical label
- D. Classification, continuous label



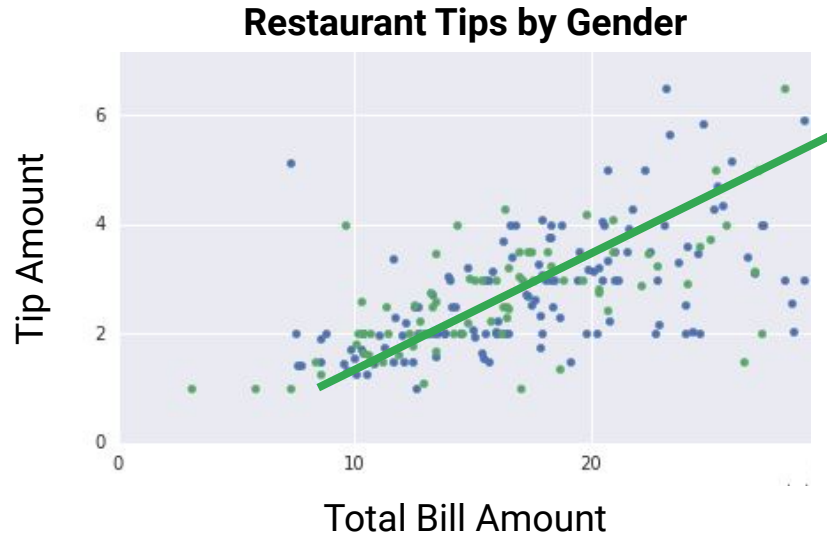
# Quiz: Supervised learning

Imagine you are in banking and you are creating an ML model for detecting if transactions are fraudulent or not. Is this classification or regression and why?

- A. Regression, categorical label
- B. Regression, continuous label
- C. Classification, categorical label
- D. Classification, continuous label

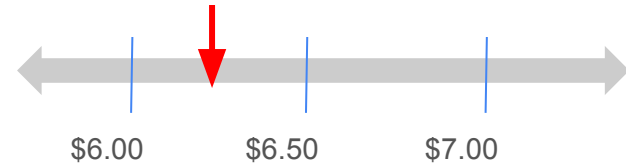


# Use regression for predicting continuous label values

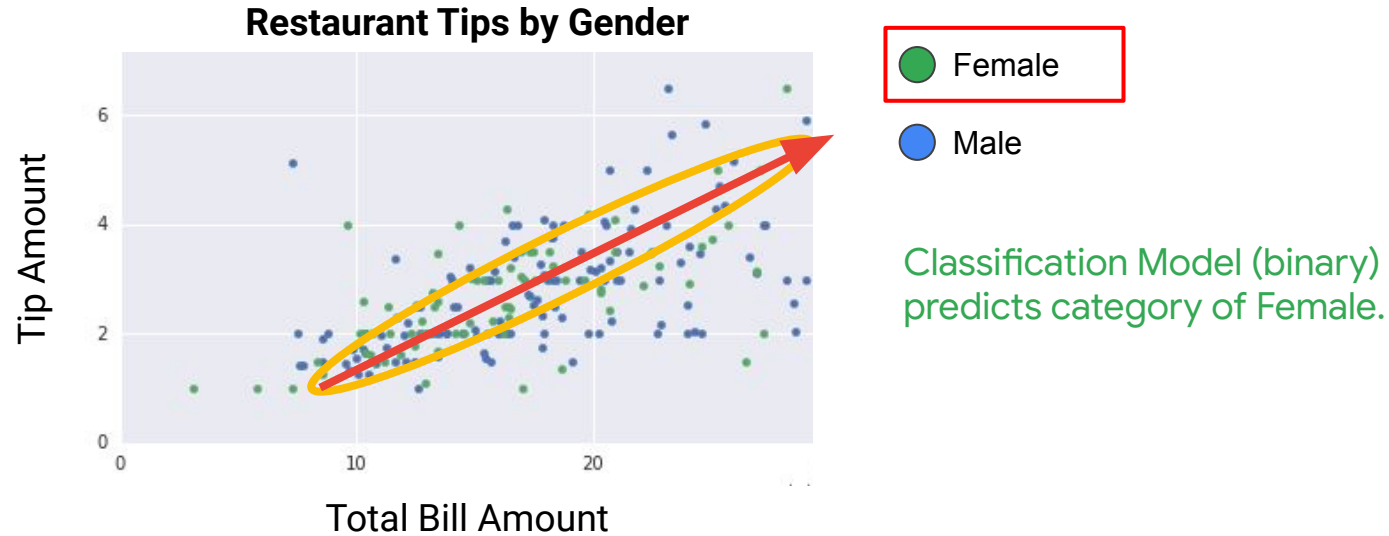


- Female
- Male

Regression Model  
(linear) predicts tip  
amount of \$6.25.



# Use classification for predicting categorical label values

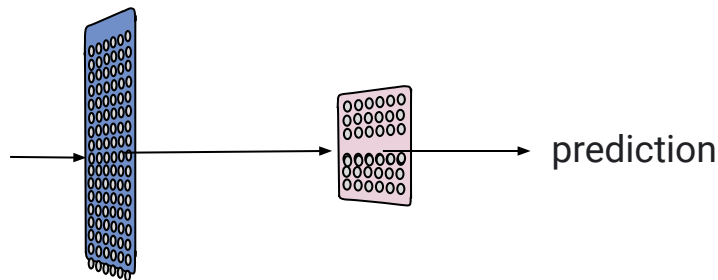


# A data warehouse can be a source of structured data training examples for your ML model

```
SELECT
  gestation_weeks,|
  mother_age,
  cigarette_use,
  alcohol_use,
  weight_gain_pounds
FROM
  `bigquery-public-data.samples.natality`
WHERE cigarette_use is not null AND alcohol_use is not null
```

weight	year	mother_age	gestation_weeks	cigarette_use	alcohol_use
7.86	2003	25	39	false	false
7.5	2003	21	39	false	false
8.06	2004	29	40	false	false
7.56	2004	38	37	false	false
7.06	2003	22	38	false	false

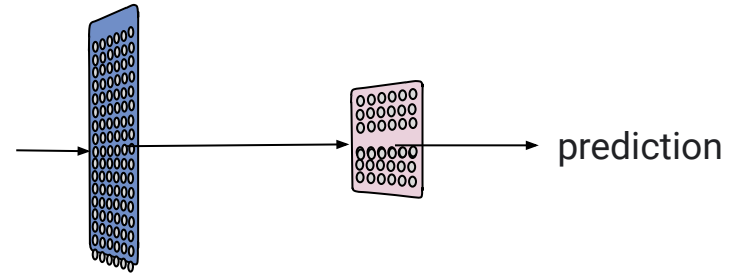
Data on births is sourced from our BigQuery Data Warehouse using SQL.



Since baby weight is a continuous value, use regression to predict

weight	year	mother_age	gestation_weeks	cigarette_use	alcohol_use
7.86	2003	25	39	false	false
7.5	2003	21	39	false	false
8.06	2004	29	40	false	false
7.56	2004	38	37	false	false
7.06	2003	22	38	false	false

Weight is stored as a floating point number, representing a continuous (real) value.



Regression DNN Model

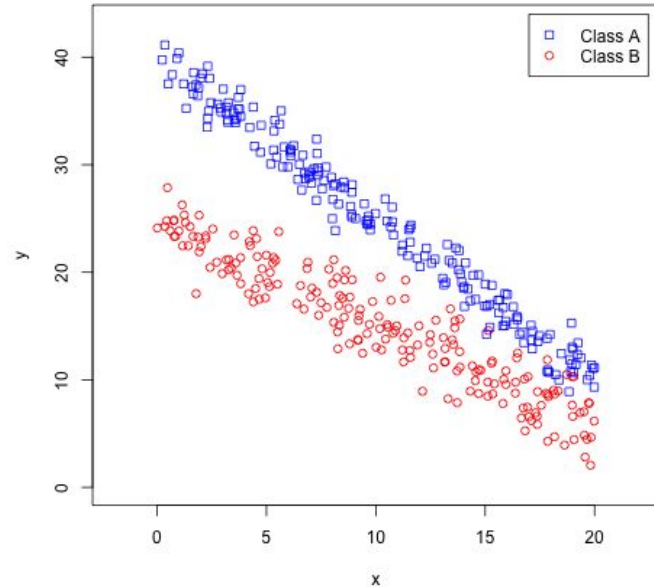




# Quiz: Regression/Classification

Is this dataset a good candidate for linear regression and/or linear classification?

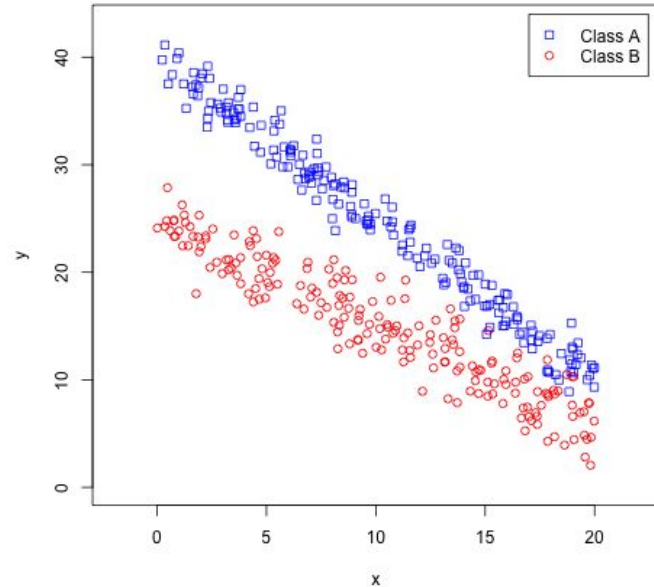
- A. Linear classification
- B. Both
- C. None of the above



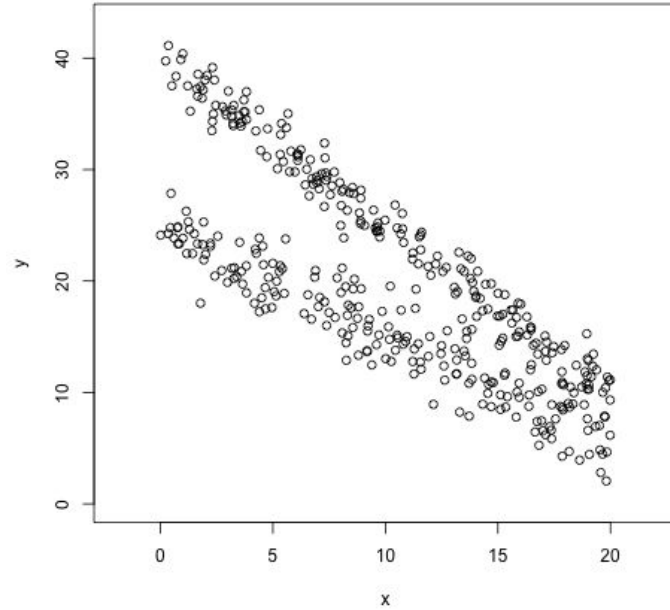
# Quiz: Regression/Classification

Is this dataset a good candidate for linear regression and/or linear classification?

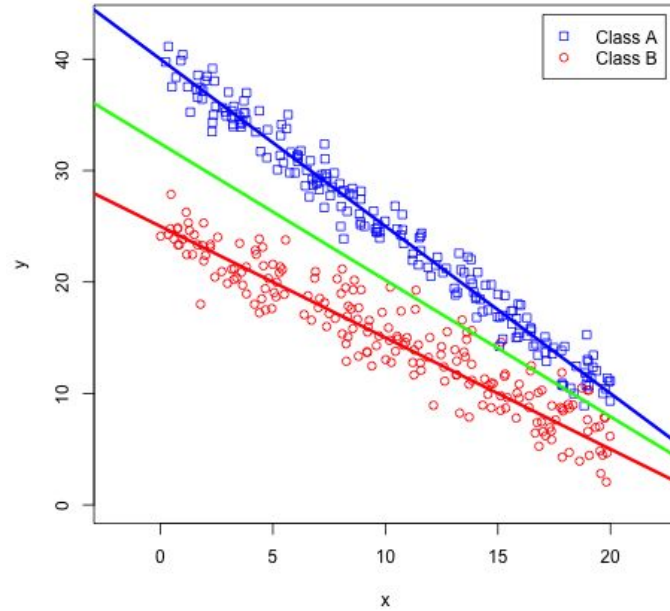
- A. Linear classification
- B. Both
- C. None of the above



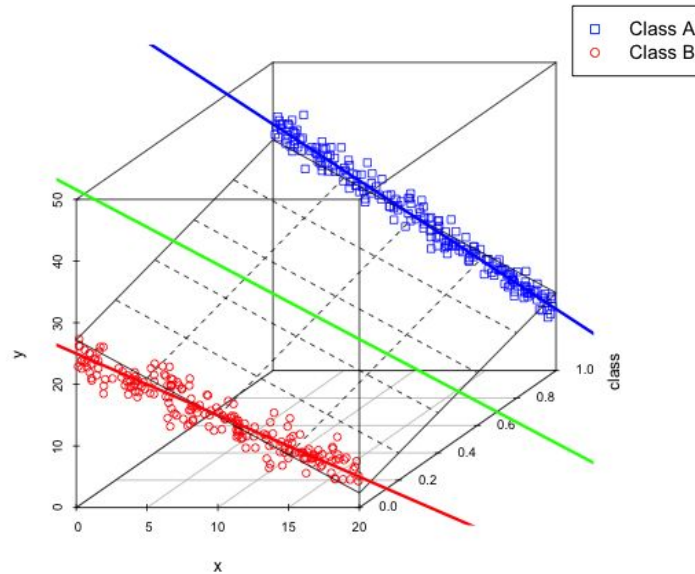
Is this dataset a good candidate for linear regression and/or linear classification?



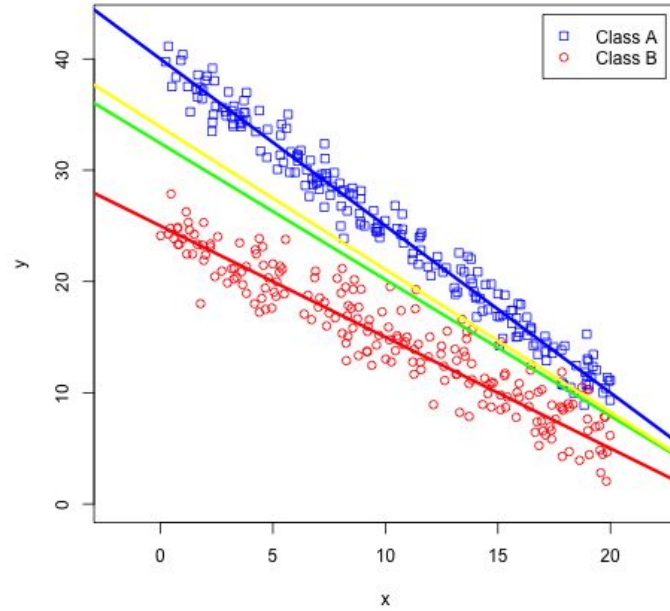
Is this dataset a good candidate for linear regression and/or linear classification?



Is this dataset a good candidate for linear regression and/or linear classification?



Is this dataset a good candidate for linear regression and/or linear classification?



# Agenda

---

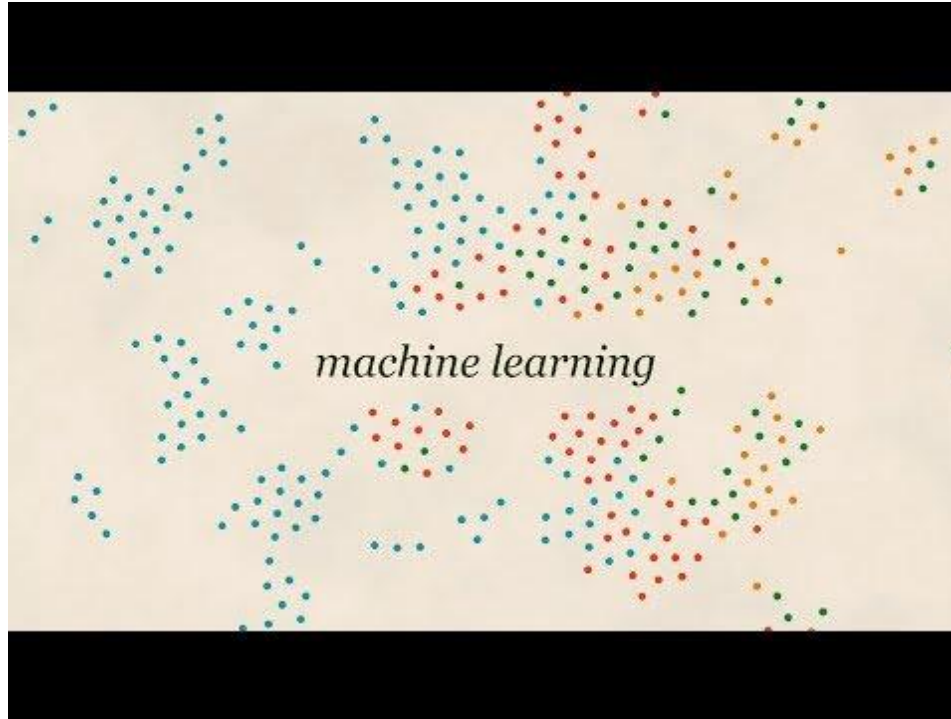
Python notebooks in the Cloud

Supervised Learning

**Inclusive ML**

Short History of ML

# Human biases lead to biases in ML models





# Unconscious biases exist in data

**Unconscious bias** from “the world” that we might reflect in ML when using existing data

Collecting data

Labeling data

**Unconscious bias** in our procedures that we might reflect in our ML

## Examples of Human Biases in Data

Reporting bias

Selection bias

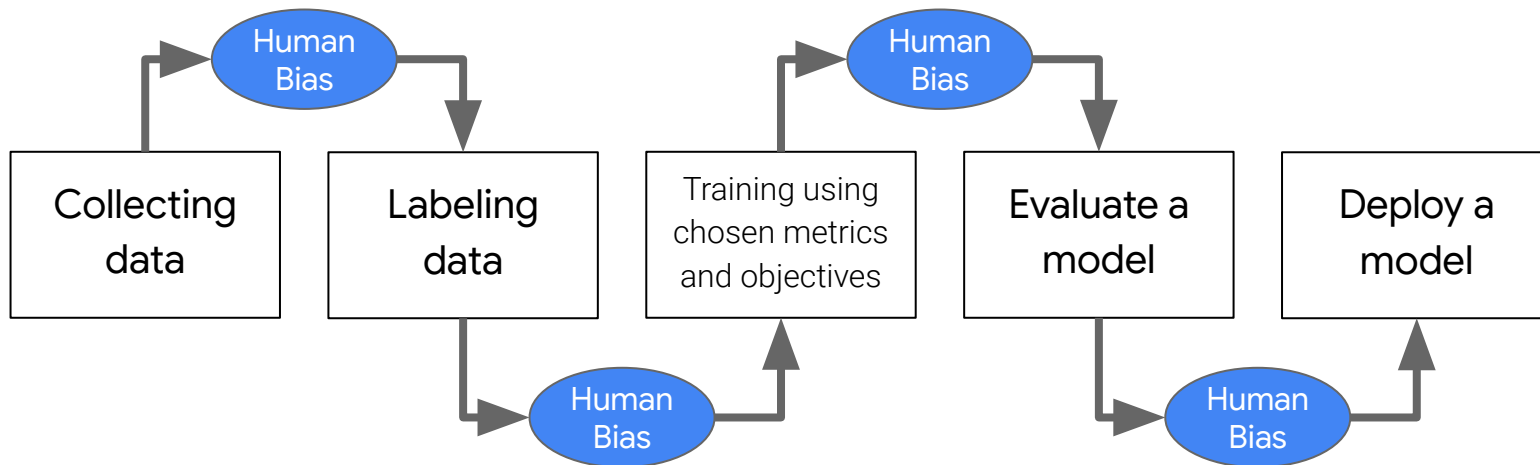
## Examples of Human Biases in Collection and Labeling

Confirmation bias

Automation bias



# A typical ML pipeline *with bias*



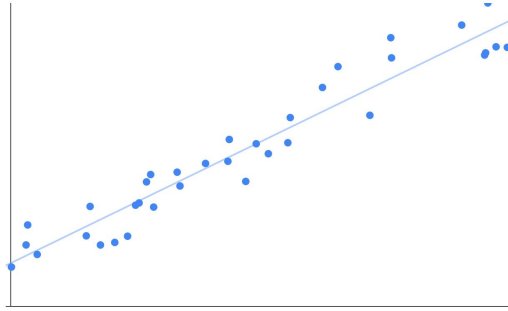
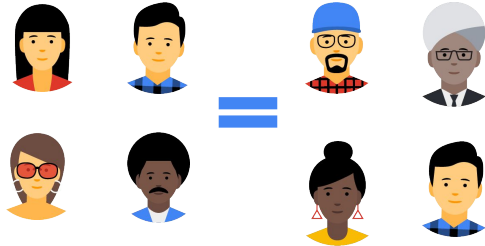
## 2 Avoid creating or reinforcing unfair bias

ML models learn from existing data collected from the real world, and so an accurate model may learn or even amplify problematic pre-existing biases in the data based on race, gender, religion, or other characteristics.

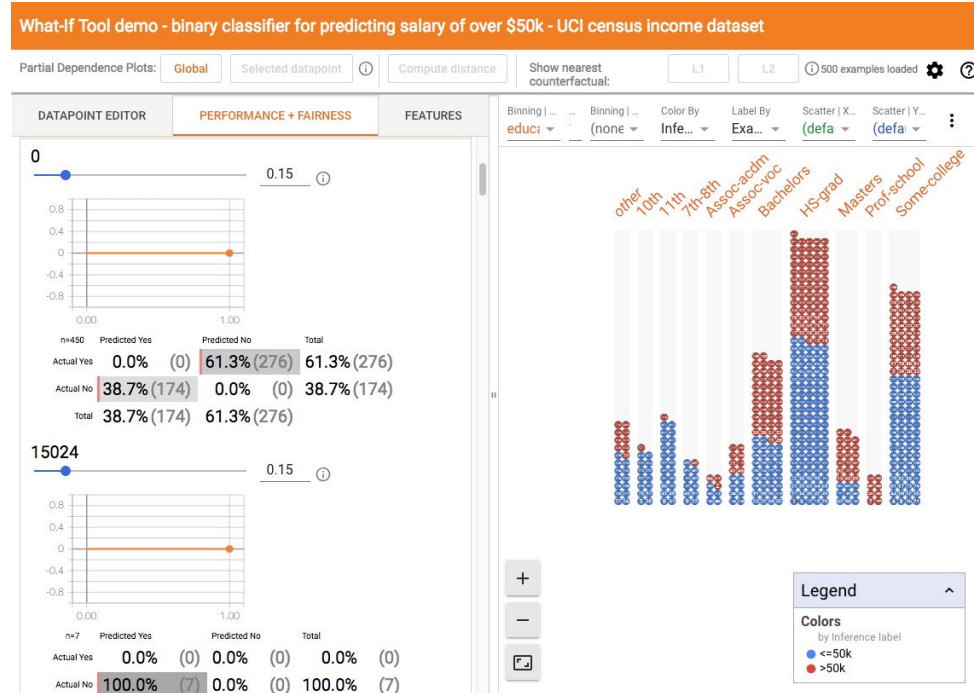
[ai.google/principles](https://ai.google/principles)



# A Checklist for Bias-Related Issues



# Tools for Responsible AI



# Agenda

---

Python notebooks in the Cloud

Supervised Learning

Inclusive ML

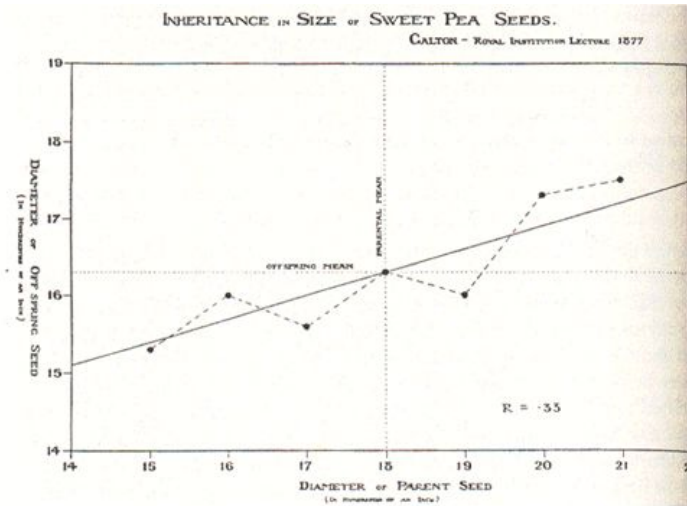
**Short History of ML**

Linear regression was invented when computations were done by hand, but it continues to work well for large datasets

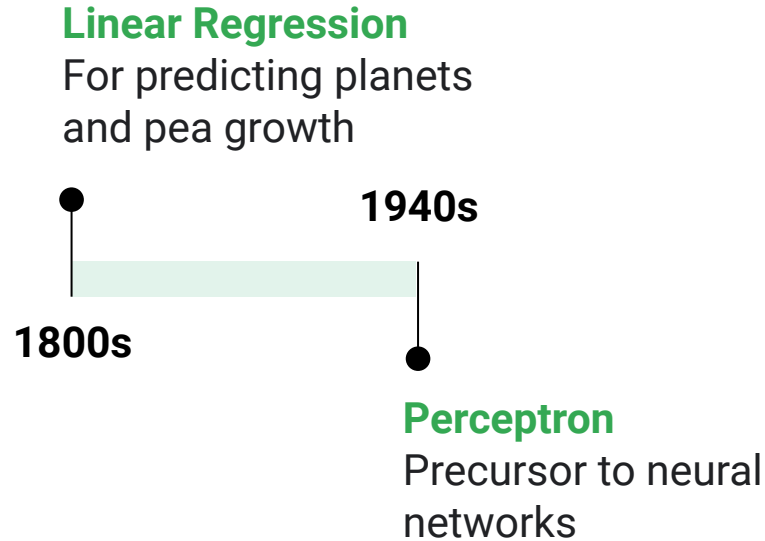
### Linear Regression

For predicting planets  
and pea growth

1800s

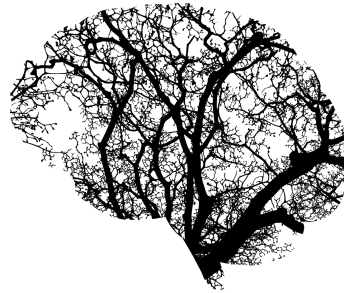
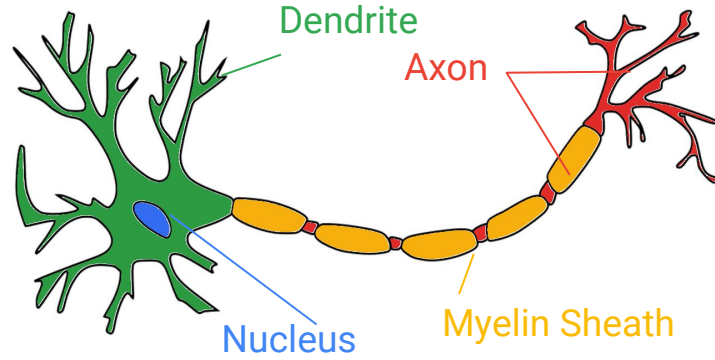


# The perceptron was a computational model of a neuron

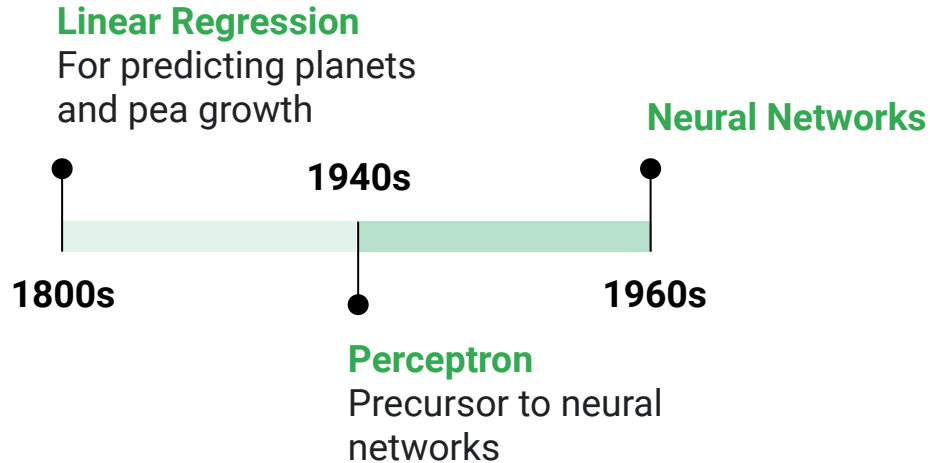




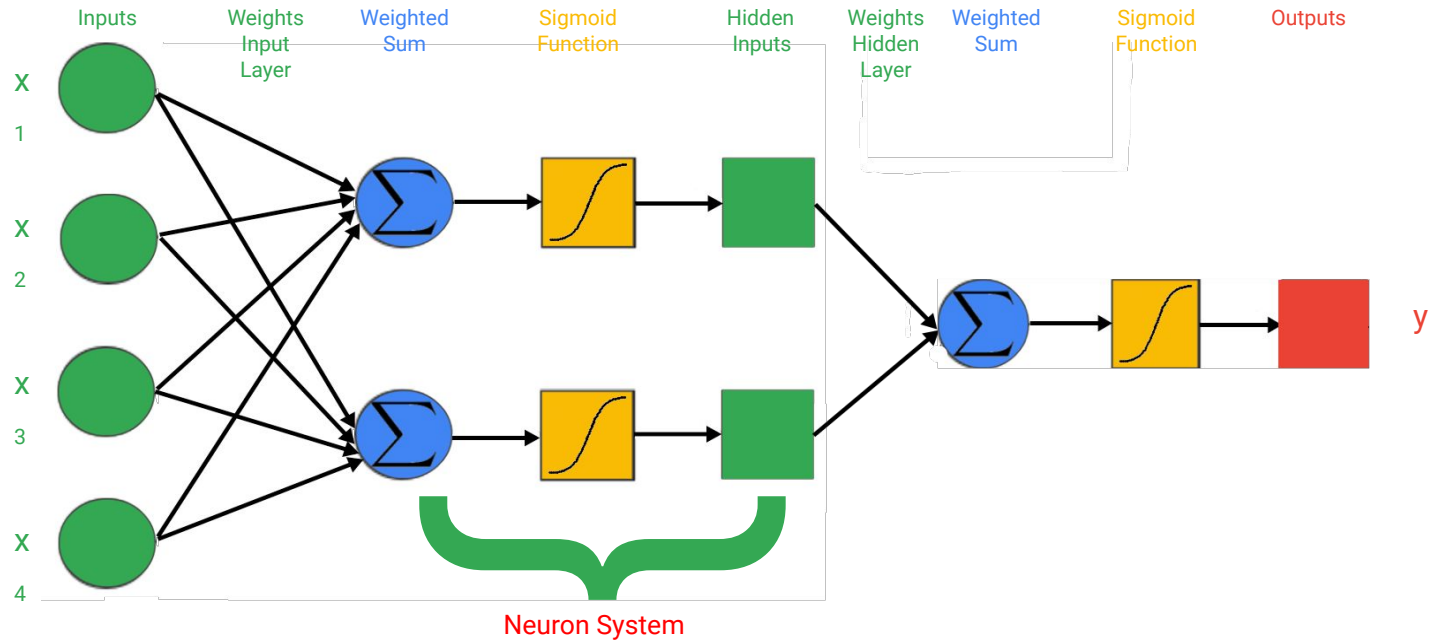
# Perceptron motivation: Neurons



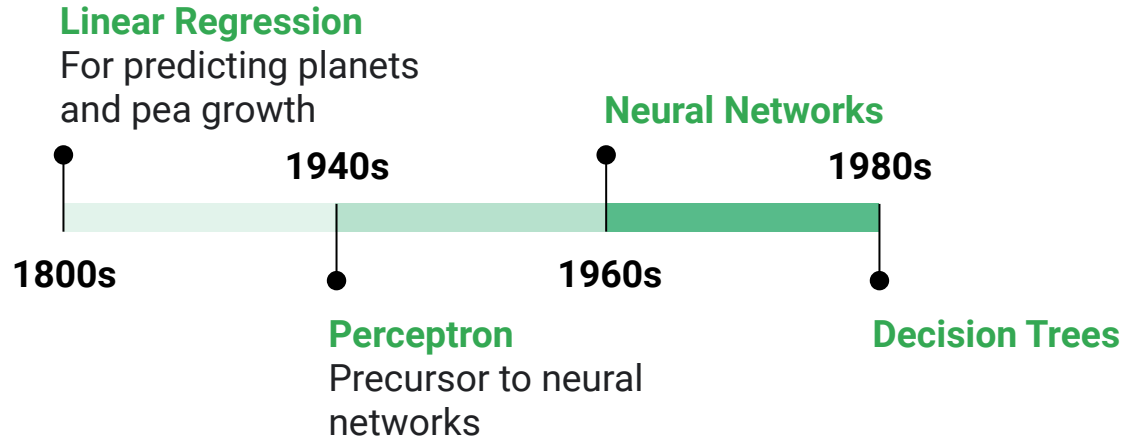
Neural networks combine layers of perceptrons, making them more powerful but also harder to train effectively



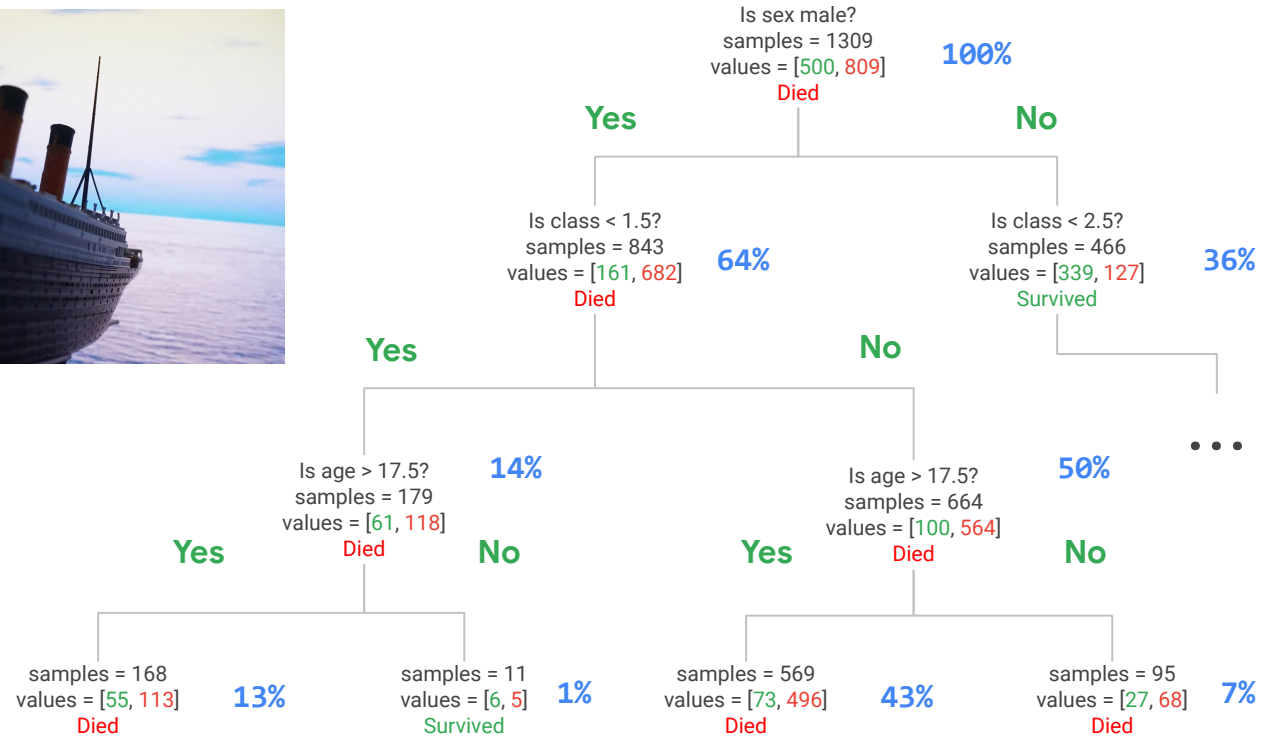
# Neural networks: Multi-layer perceptron



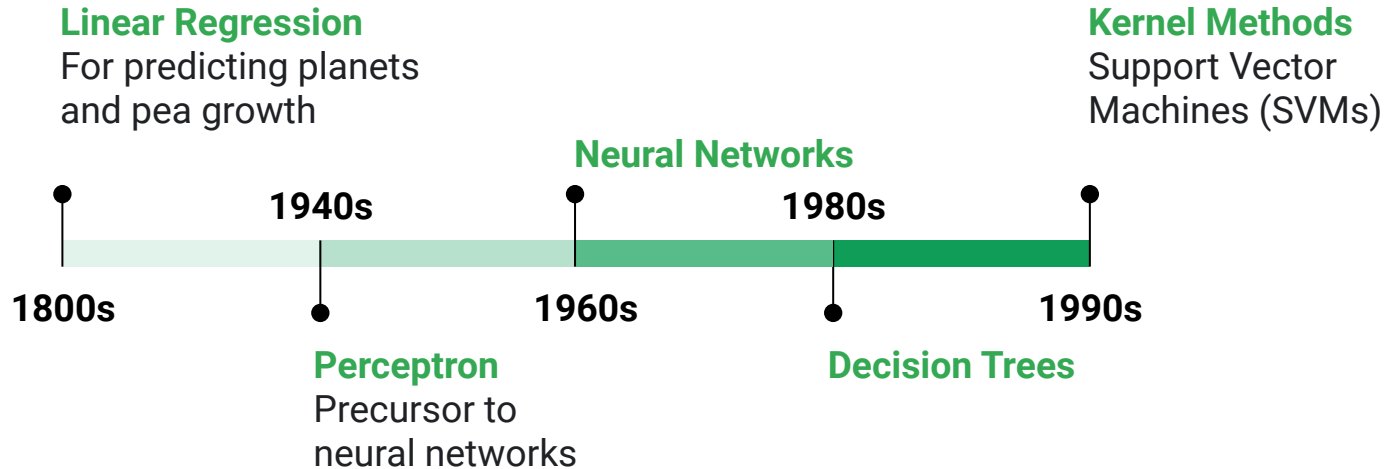
Decision trees build piecewise linear decision boundaries, are easy to train, and are easy for humans to interpret



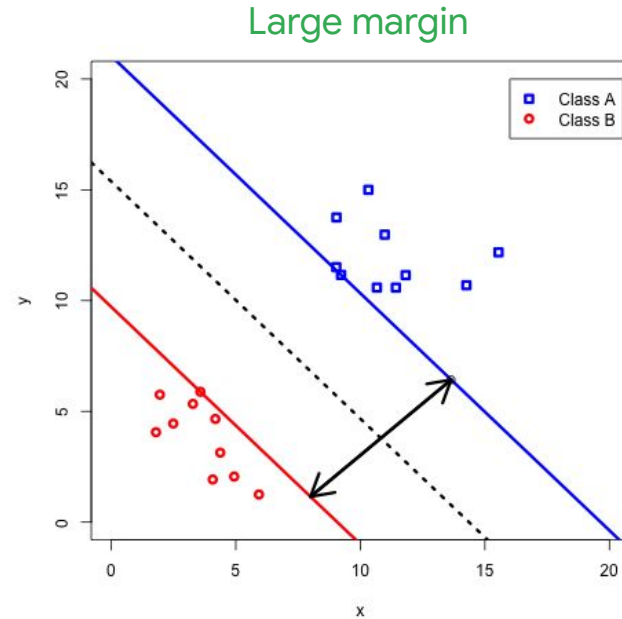
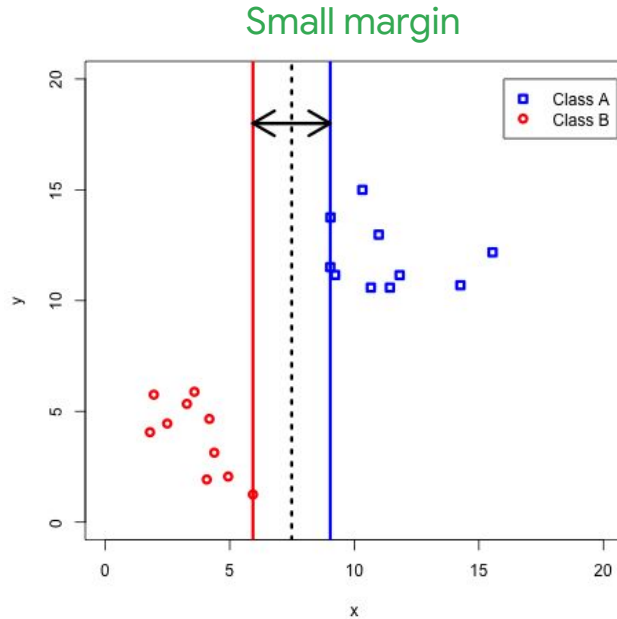
# Decision trees and the Titanic



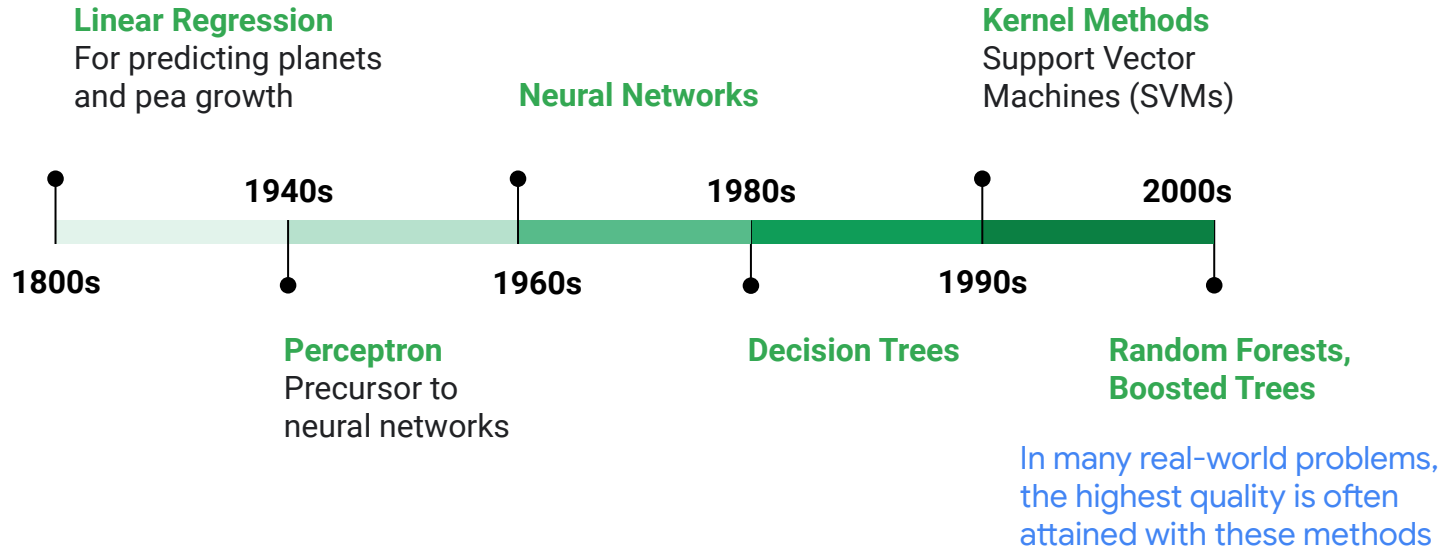
Support vector machines are nonlinear models that build maximum marginal boundaries in hyperspace



# SVMs maximize the margin between two classes

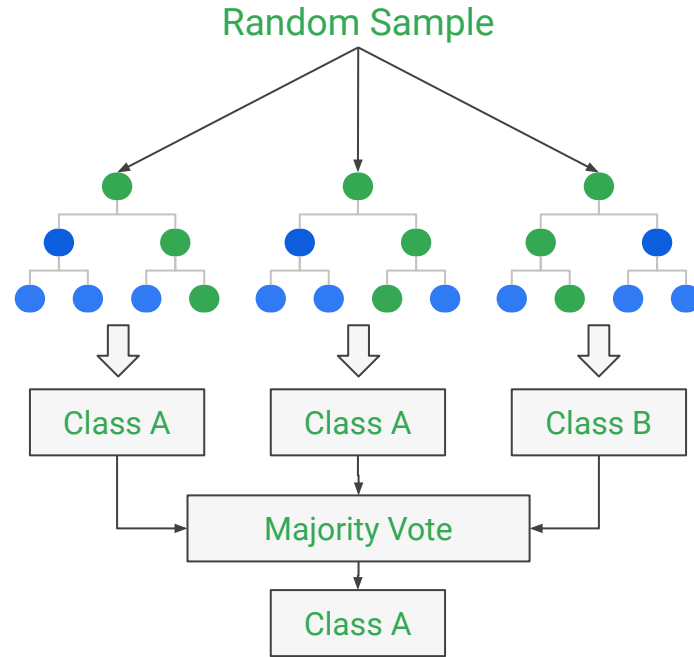


# Random forests, bagging, and boosting are very effective predictors built by combining lots of very simple predictors

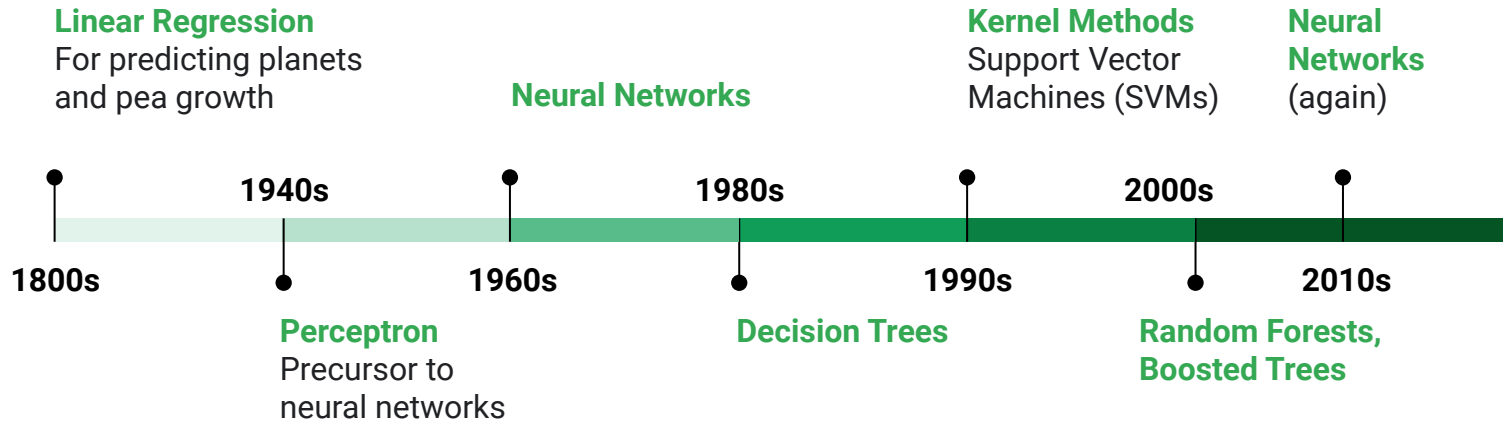




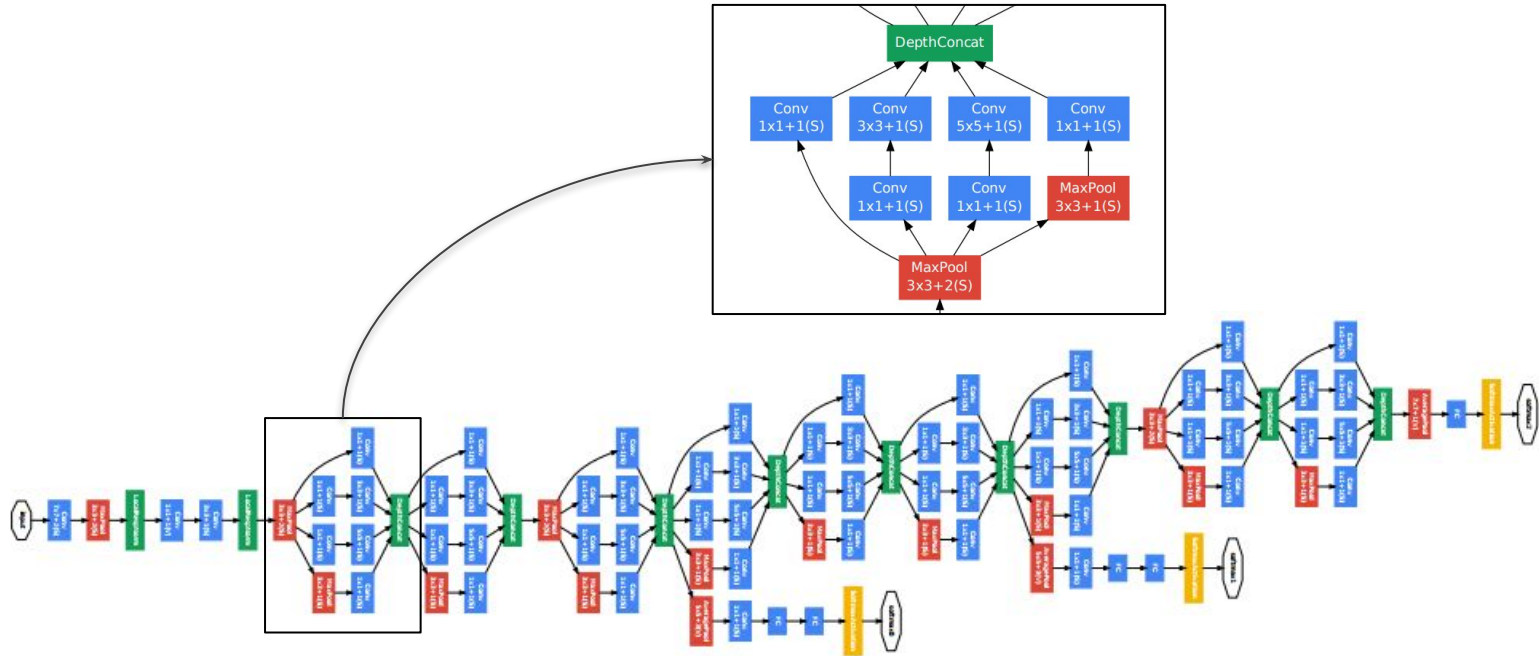
# Random forest: Strong learner from many weak learners



With the advantage of technical improvements, more data, and computational power, neural networks made a comeback



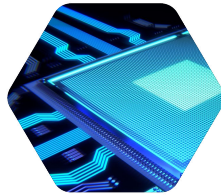
# Inception/GoogleLeNet Deep Neural Network



# Neural networks are outperforming most other approaches in many domains



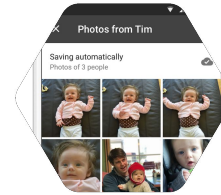
Large  
amounts  
of data



Available  
Computational  
Power



Available  
Infrastructure



Tasks and  
Goals we care  
about

Note that there are no models that are  
universally better, they're just different.

