gTech
<professional services/>

WOMEN IN DATA SCIENCE
@ GOOGLE

kaggle

WiDS@Google Datathon Workshop

WOMEN IN DATA SCIENCE

- A **conference** with 150+ regional events worldwide in more than 60 countries, reaching 100,000 participants annually.
- A **datathon**, encouraging participants to hone their skills using a social impact challenge.
- A **podcast** series, featuring data science leaders from around the world talking about their work, their journeys, and lessons learned along the way.
- An **education outreach** program to encourage secondary school students to consider careers in data science, artificial intelligence (AI), and related fields.
- A **workshop** series to build your data science skills, inspiring women and girls with role model instructors.

Agenda (US East Coast & EMEA time zone)

- **9:20 -10:05 EST / 14:20 - 15:05 GMT**

  Introduction to Machine Learning
- **10:15 - 11:05 EST / 15:15 - 16:15 GMT**

  Python code walkthrough for datathon forecasting problem
- **11:15 - 11:30 EST / 16:15 - 16:30 GMT**

  Datathon challenge introduction & team formation for hands-on exercise
- **11:30 - 12:30 EST / 16:30 - 17:30 GMT**

  Datathon Challenge hands-on exercise with mentor support

Agenda (US West Coast & APAC time zone)

- **8:20 -9:05 GMT+9 / 16:20 - 17:05 PST**

  Introduction to Machine Learning
- **9:15 - 10:05 GMT+9 / 17:15 - 18:05 PST**

  Python code walkthrough for the datathon forecasting problem
- **10:15 - 10:30 GMT+9 / 18:15 - 18:30 PST**

  Datathon challenge introduction & team formation for hands-on exercise
- **10:30 - 11:30 GMT+9 / 18:30 - 19:30 PST**

  Datathon Challenge hands-on exercise with mentor support

Introduction to Machine Learning

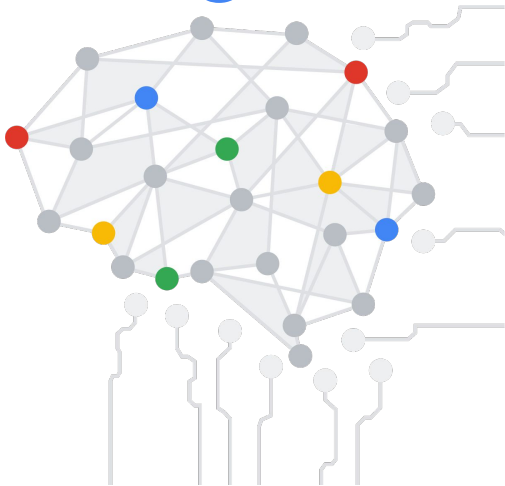
gTech
<professional services/>

WiDS Datathon

Introduction to Machine Learning

gTech gPS Data Science
Christiane Ahlheim, Min Baek

Feb 2023



Housekeeping

Please stay muted

We have a lot to cover – please put questions in the chat and if there's time at the end, we'll come back to those
You can always bring questions to your mentors, too

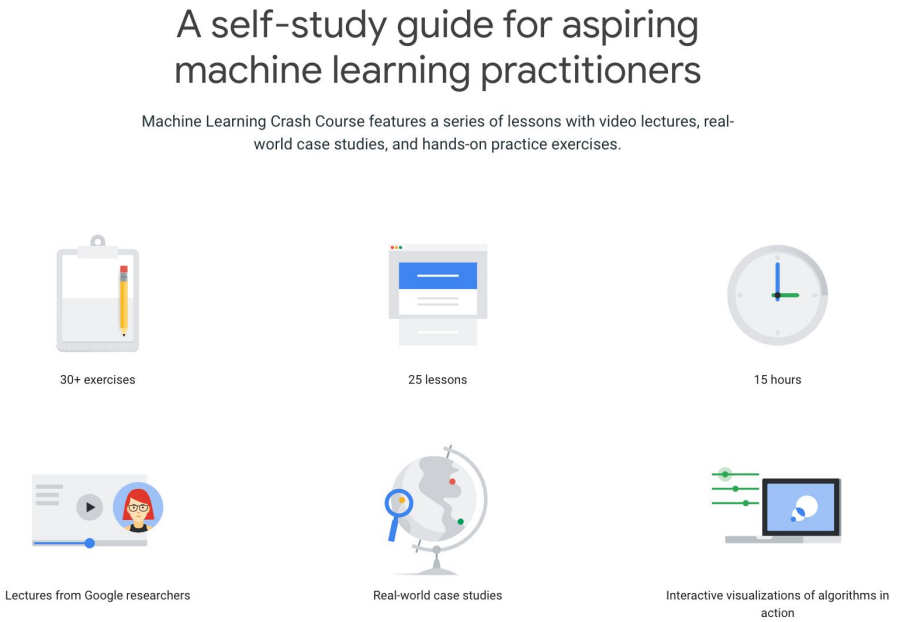
What we'll cover in the next 45 minutes

- What is Machine Learning?
- Common distinctions: Supervised vs Unsupervised
- Model Generalization
- Supervised Learning
  - Classification
  - Regression

For more details...

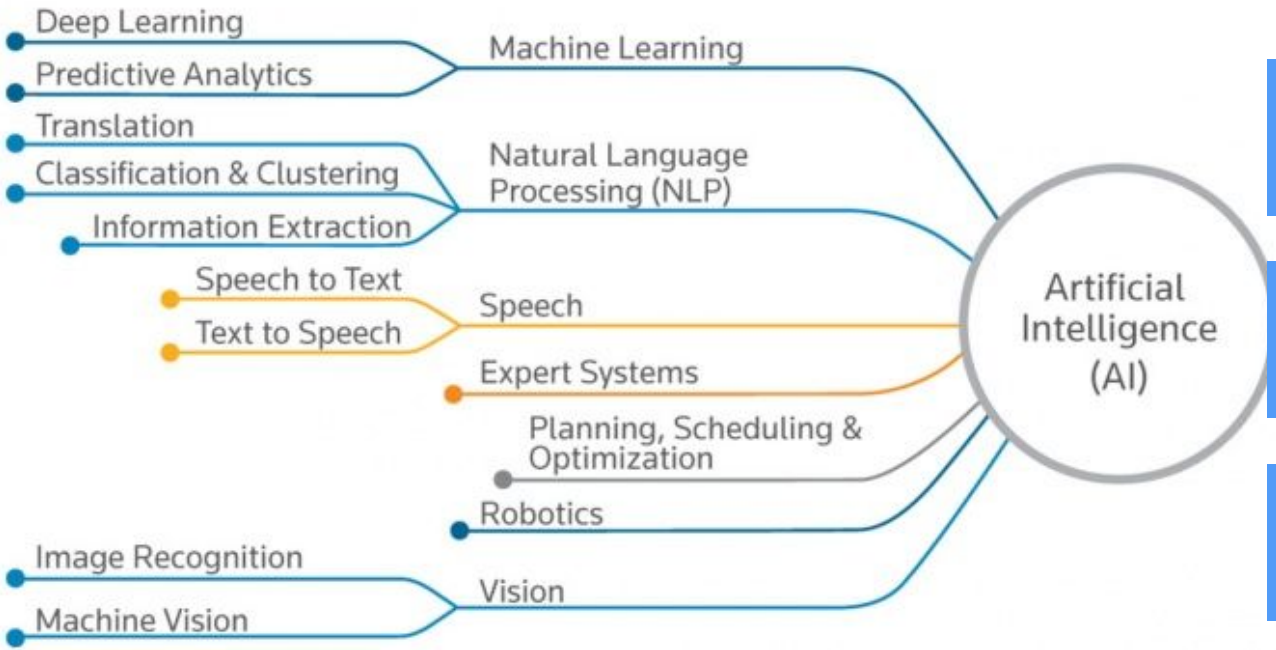
[Machine Learning Crash Course | Google Developers]

Source of most of the content shared here.



Common Terminology

Machine Learning is...



One branch of the field of Artificial Intelligence

A way of solving problems without explicitly codifying the solution

A way of building systems that improve themselves over time

Source: Neota Logic

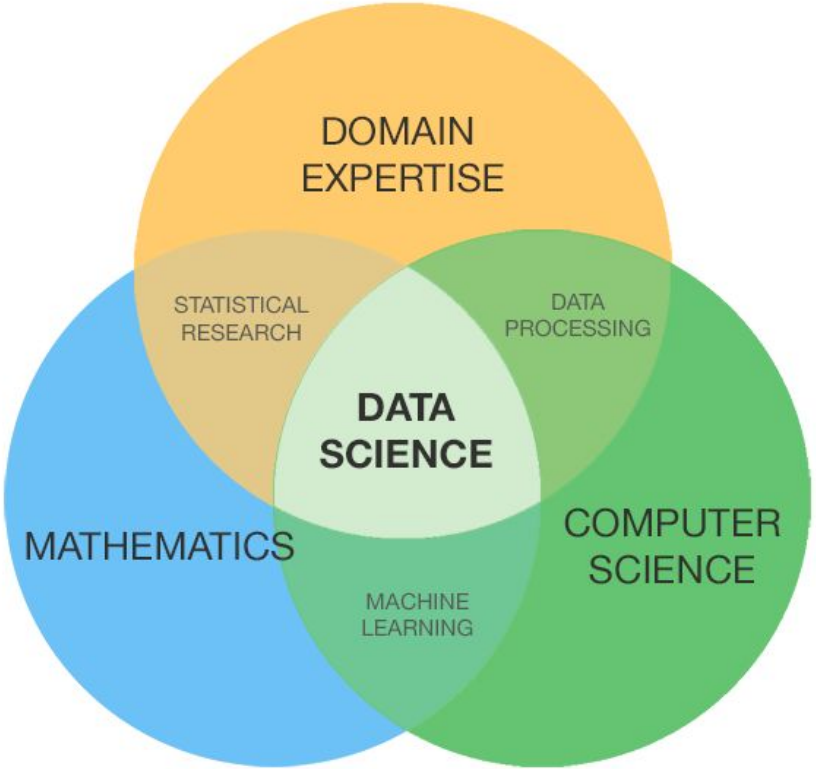
Machine Learning ⇄ Data Science

Data Science:

- Solving business problems in a data-driven way
- Include define problem statement, data processing and model building

Machine Learning:

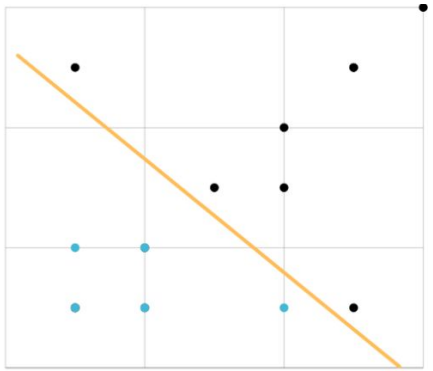
- A practice of using algorithms to capture the insights from big data
- One of the tools that Data Scientist uses



Source: Palmer, Shelly. Data Science for the C-Suite. New York: Digital Living Press, 2015. Print.

Supervised Learning

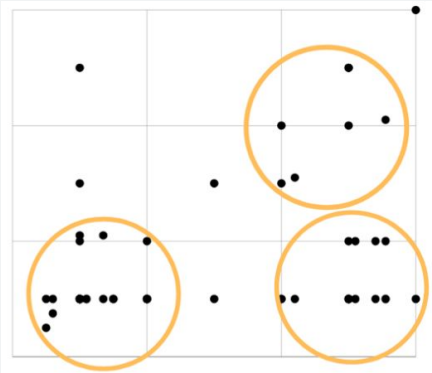
- Supervised learning is the machine learning task that **use labeled datasets** to train algorithms which will classify data or predict outcome



- **Classical examples:**
  - Time Series Forecasting: Stock price, Sales forecast
  - Classification: Handwriting Recognition, Tumor Detection
  - Regression: House rent, Car price prediction

Unsupervised Learning

- Unsupervised learning is the type of algorithm that learn pattern from **untagged** data



- **Classical examples:**
  - Customer segmentation
  - Feature reductions

This year's WiDS datathon

> [...] predict the energy consumption using building characteristics and climate and weather variables .

Model Generalization

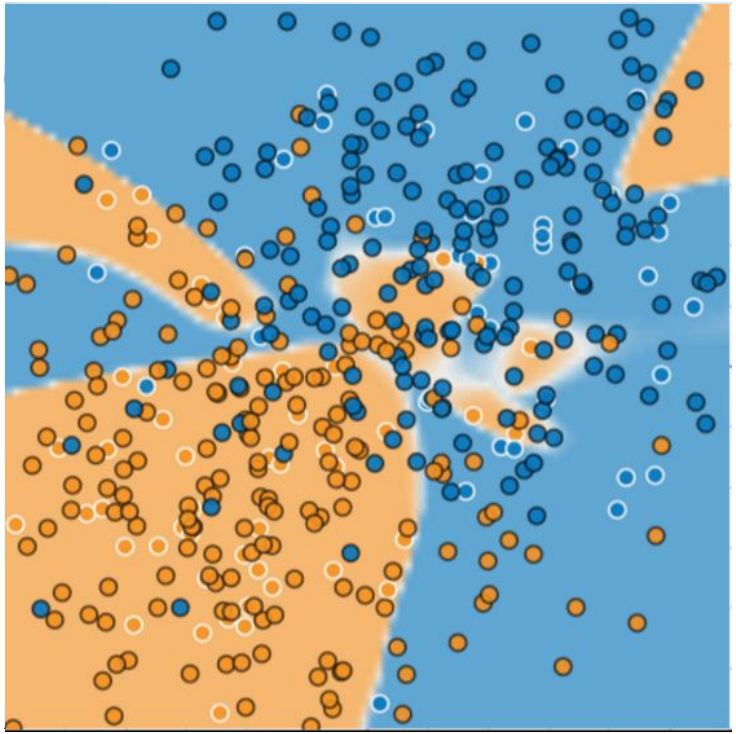
Generalization: Over- and Underfitting

The goal for each ML algorithm: predict well on **new data.**

Risk: (Complex) models can **overfit** peculiarities in your data, instead of learning the true signals.

This results in **poor performance** on new data points.

Source: [Generalization: Peril of Overfitting | Machine Learning Crash Course | Google Developers]

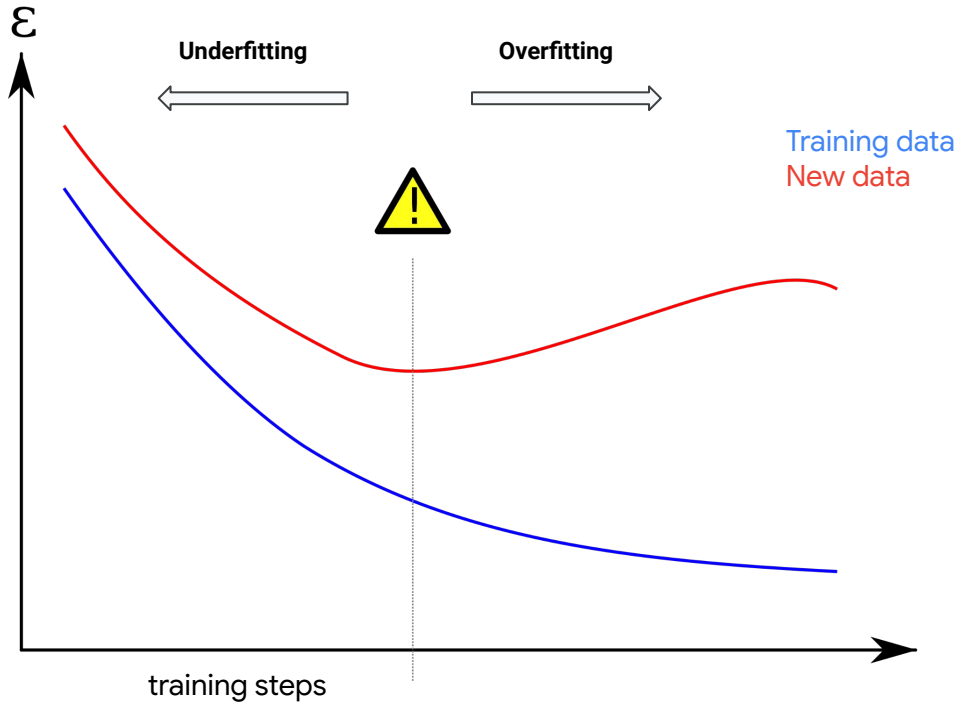


Generalization: Over- and Underfitting

We can diagnose over- and underfitting by inspecting the model performance on our training data (blue) and new data (red).

Overfitting: The error on the training data decreases, but *increases* on the new data

Underfitting: The error on the training data is still too high and could go down further.



By Gringer - Own work, CC BY 3.0, https://commons.wikimedia.org/w/index.php?curid=2959742

Generalization: Training- and Test-Set

How can we know how our model will perform on new data points?

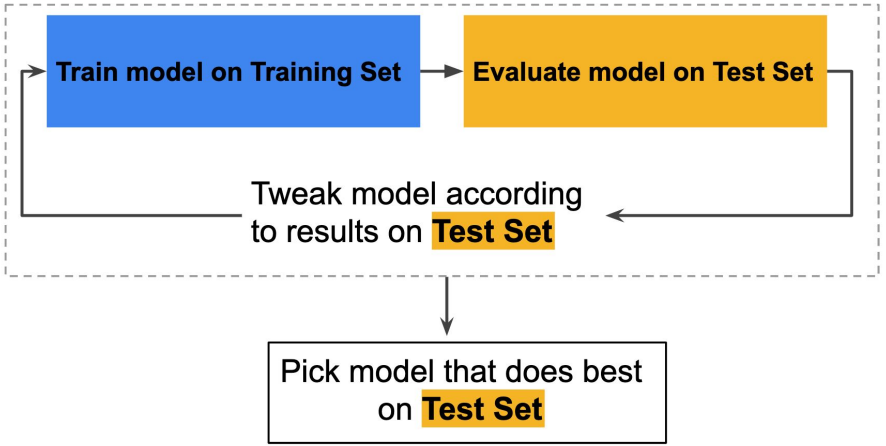
We split the data!

The test set needs to:

- Be large enough to yield statistically meaningful results
- Be representative of the whole dataset
- Be independent of the training data

Never train on test data! If your model performance is too good, check that the training data has not leaked into the test data.

Rule of thumb: 80/20 split

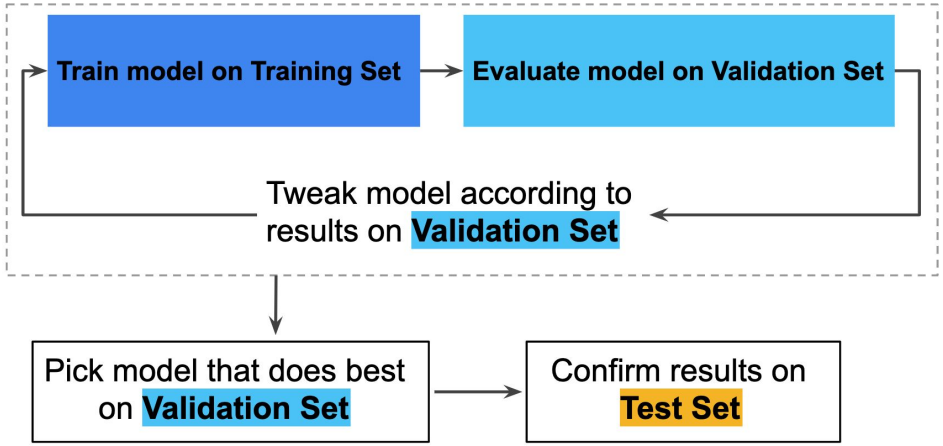


Generalization: Validation Set

Introducing a test-set already reduces the risk of overfitting greatly, but we still risk overfitting to the *test set*.

This is why general best practice is to have three splits: training, validation, and test set.

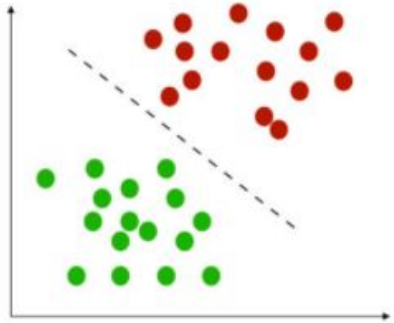
In this workflow, only the final model is checked against the test set, and risks of overfitting are thus reduced further.



Classification and Regression

Classification

- Labels are **categorical**, which can be two (binary) or more (multiclass)

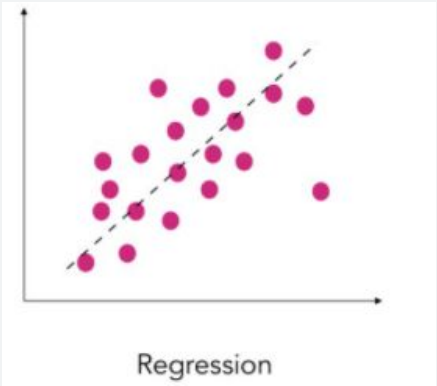


Classification

- Classification model predict each observation's category
  - Output the probability for each category
- Classical examples:
  - Tumor detection
  - Handwriting recognition
  - ...

Regression

- Labels are (usually) **continuous**, but could, e.g., only be integers



Regression

- Regression model predict each observation's value
  - Output the actual value as prediction
- Classical examples:
  - Stock market
  - Sales
  - ...

By Maël Fabien,
https://maelfabien.github.io/machinelearning/ml_base/#supervised-vs-unsupervised

This year's WiDS datathon

> [...] predict the energy consumption using building characteristics and climate and weather variables .

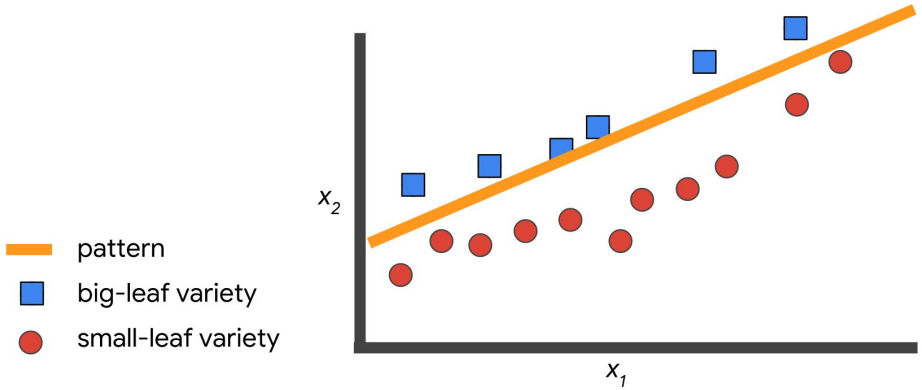
Classification Deep-Dive

Classification Problems

Classification: predicting categorical labels (e.g. plant type, hair color, image category)

Easiest case: binary classification, with only two labels (e.g., cat vs dog)

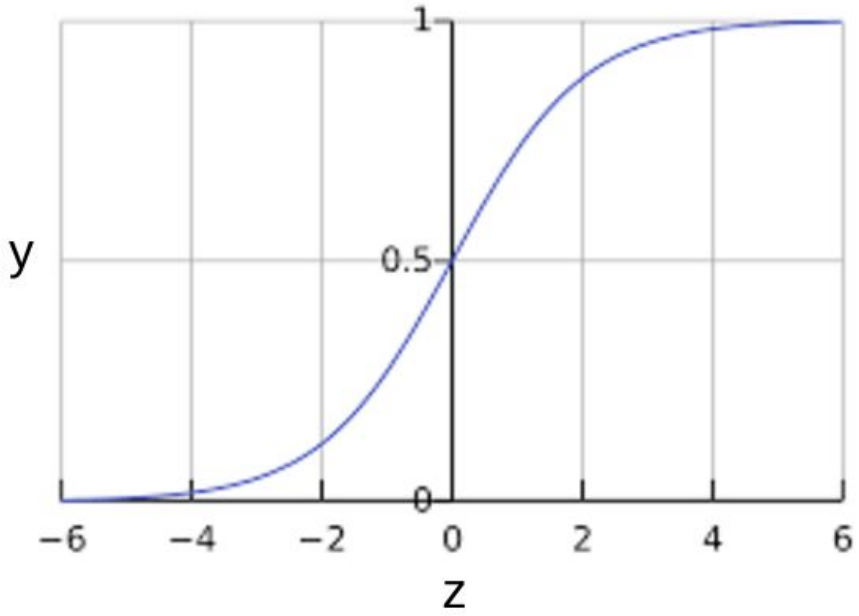
Output: predicted (probability of) label → probabilities are turned into label-predictions via **thresholding**



Example Algorithms

Logistic Regression:
Supports binary and multiclass classification

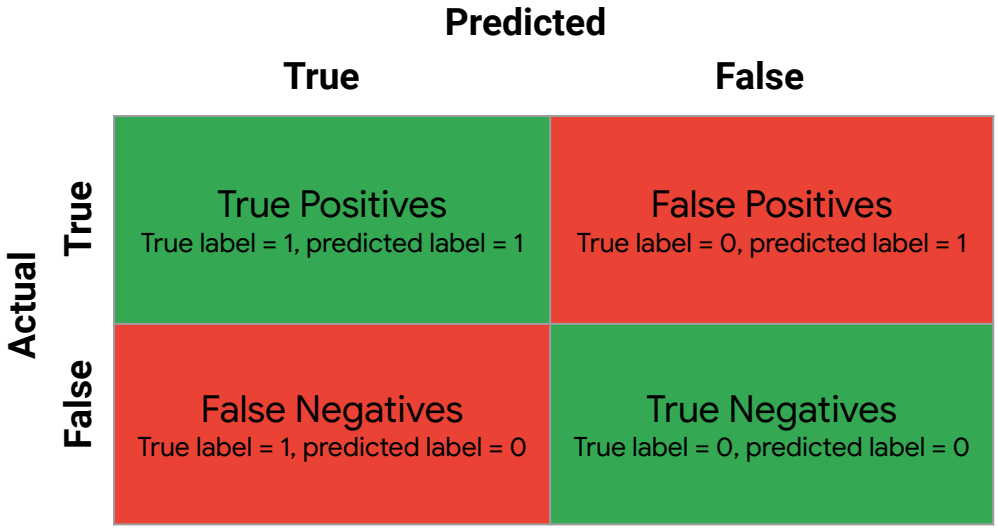
Tree-based models:
Also support regression (see next section), range from
Decision Trees to
Random Forests
and gradient-boosted Trees like **LightGBM.**



Model performance: Confusion Matrix

Ideally, we want high values in the green cells and low values in the red cells.

But: often, we have consider trade-offs between those four outcomes.

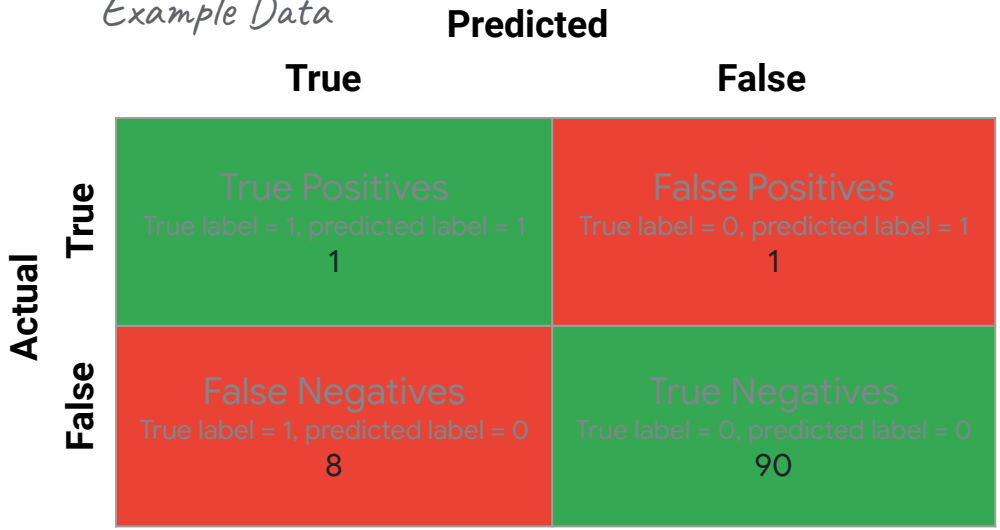


Model Performance: Accuracy

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Example Data



$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{1 + 90}{1 + 90 + 1 + 8} = 0.91$$

*Which problem could we have with **Accuracy** as a metric?*

Model Performance: Precision and Recall

Precision

What proportion of positive identifications was actually correct?

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall

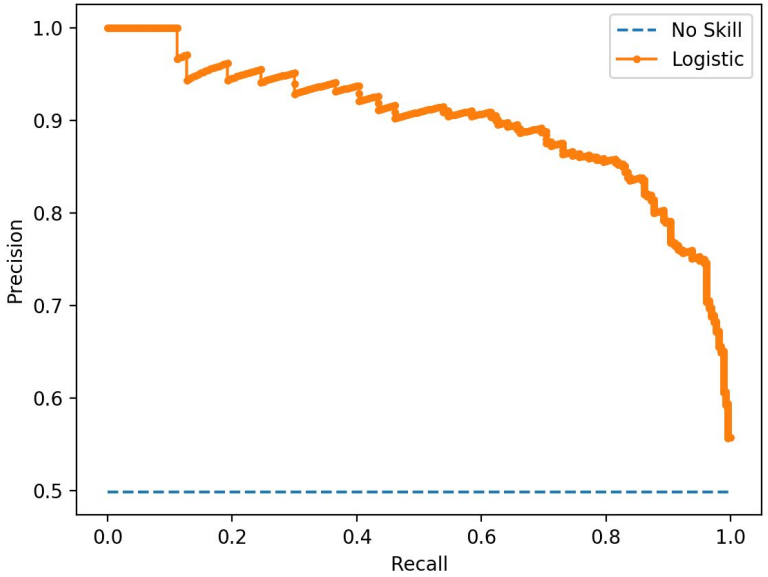
What proportion of actual positives was identified correctly?

$$\text{Recall} = \frac{TP}{TP + FN}$$

Model Performance: Precision and Recall

Both metrics need to be examined to fully evaluate the effectiveness of a model.

Usually, they are in tension: improving precision reduces recall and vice versa.



https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-imbalanced-classification/

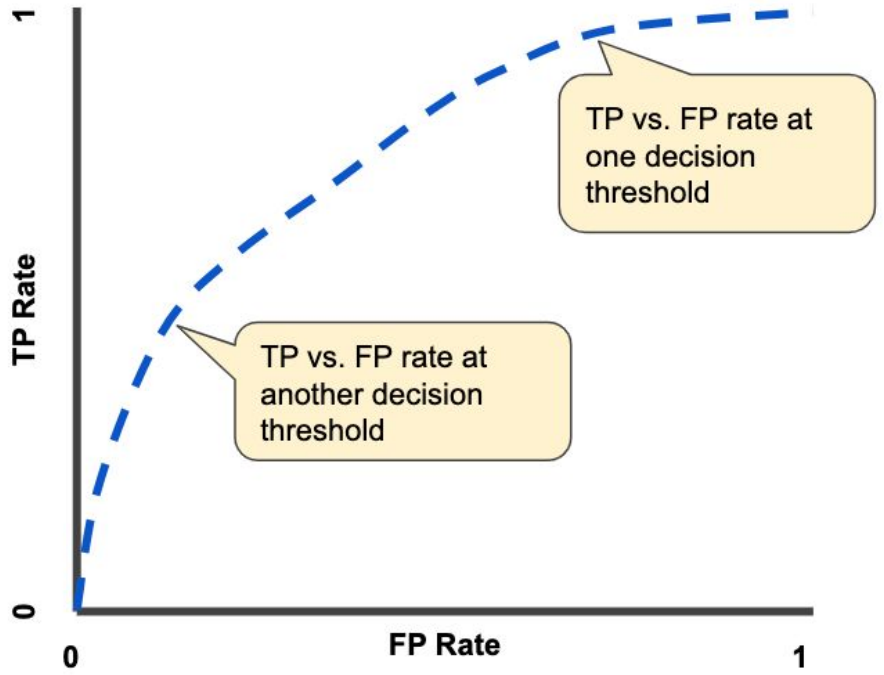
Model Performance: ROC curve and AUC

Receiver operator characteristic (ROC) curve:
Performance of a classification model at all classification thresholds, by plotting **True Positive Rate** (TPR) and **False Positive Rate** (FPR)

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

Area under the ROC Curve (AUC)
measures the **entire two-dimensional area** underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1)



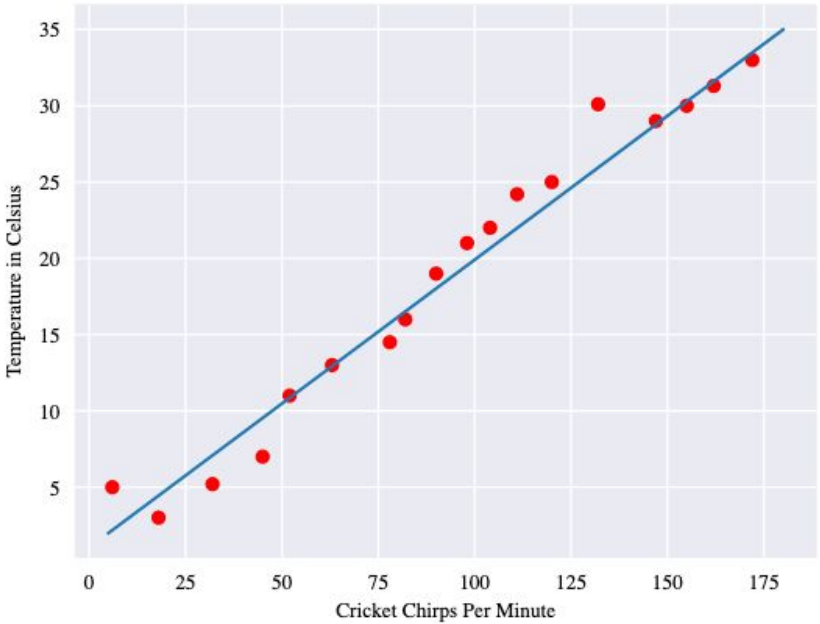
Regression Deep-Dive

Regression Problems

Regression: predicting continuous target values (e.g. temperature, costs, height)

Can be formulated as **linear** or **non-linear** models

Output: (usually) predicted **target values**



Common Algorithms

Linear Regression

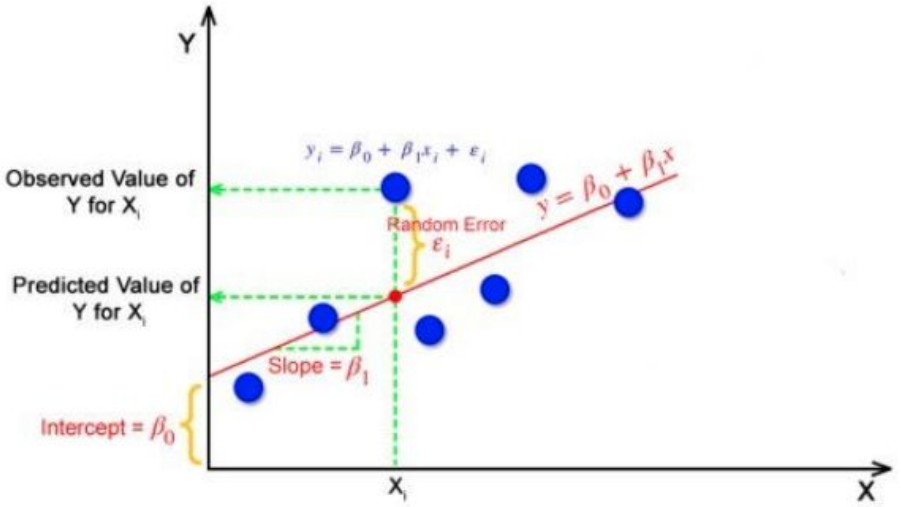
Estimate target value with a linear function of intercept and other predictors

Tree based models

Random forest regression, Gradient boosted regression

Neural Networks

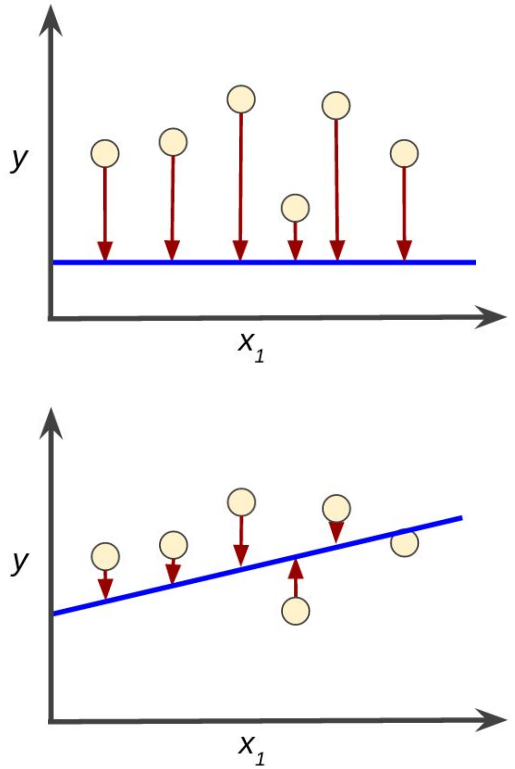
Deep Neural Network: Train a network with multiple hidden (transformation) layers to predict target value



Model Performance | Minimizing Loss

Goal: find model parameters so that predicted values are most similar to actual values, i.e. that **minimize the loss**.

$$MSE = \frac{1}{N} \sum_{(x,y) \in D} (y - prediction(x))^2$$

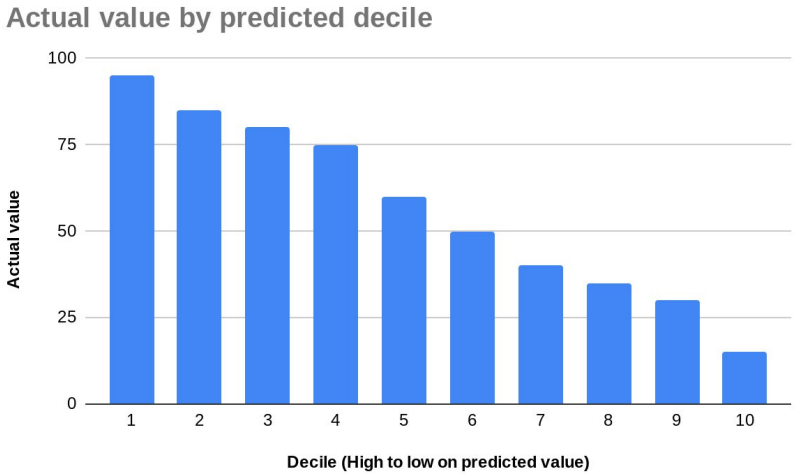


The arrows represent loss.
The blue lines represent predictions.

Model Performance | Other Metrics

Decile Lift Chart:

Average of actual value within each predicted decile



Mean Average Percentage of Error:

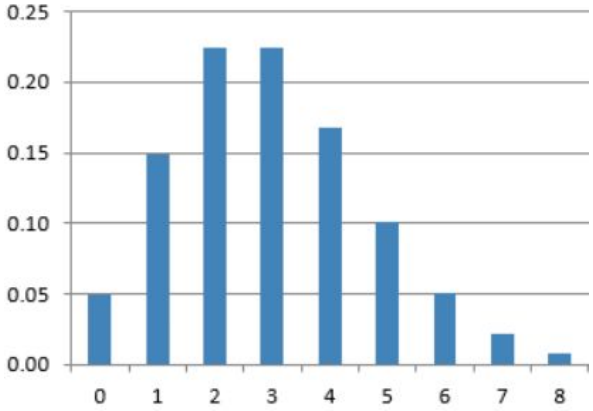
$$\text{MAPE} = \frac{100\ \%}{n} \sum_{t=1}^{n} \left| \frac{Actual\ value - Predicted\ value}{Actual\ value} \right|$$

Measure of prediction accuracy in forecasting model

Special Regression Cases

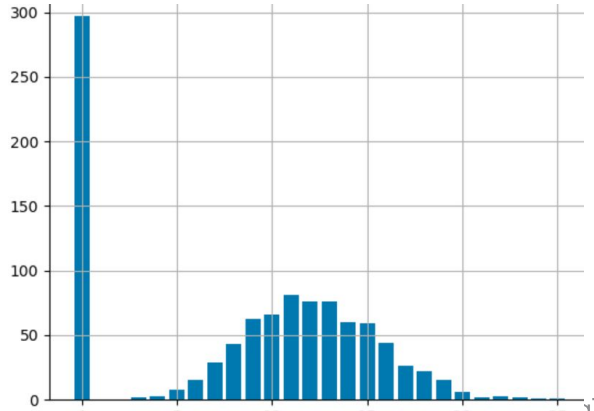
Poisson regression:

Poisson regression is applied when response variable are count data
Example: # of ER visits, # of car accident each year



Tweedie Loss/Zero-inflation regression:

Zero-inflated model is applied when you data contain excess zero-count data



Thank you!

Questions?

Colab Demo on this year's Datathon problem

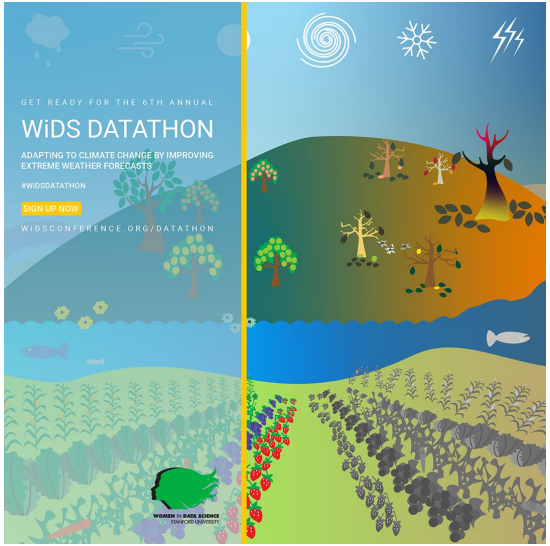
(We will go over the problem on colab notebook)

Introduction to Datathon

WiDS 2023 Datathon Challenge on Kaggle

Improve extreme weather forecast to adapt to climate change

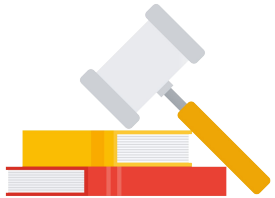
- The goal of the challenge is to improve longer-range weather forecasts to help people prepare and adapt to extreme weather events caused by climate change.
- Each row in the data corresponds to a single location (in the US) and a single start date for the two-week period.
- **Target** : Predict arithmetic mean of max and min temperature over the next 14 days.
- **Features**: Temperature, Global & US precipitation, Sea surface temperature and sea ice concentration, Geopotential height, zonal wind, and longitudinal wind, etc.
- **Evaluation Criteria** : RMSE.



WiDS Datathon
Hands-on exercise

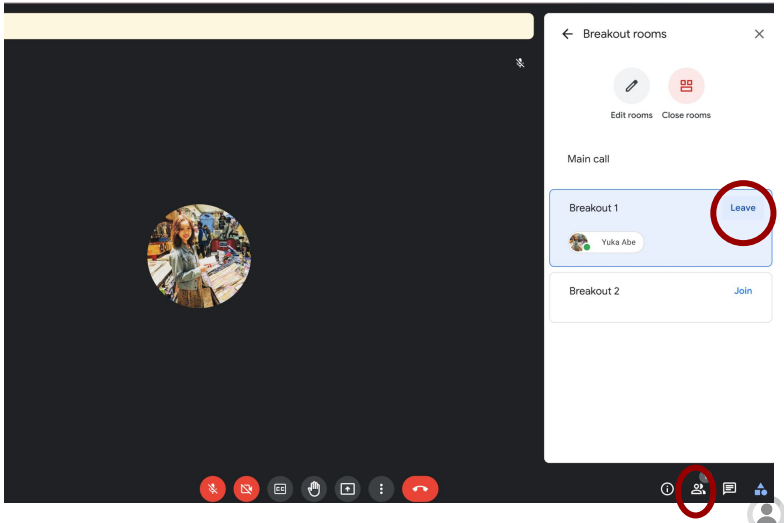
Important Datathon Rules

- The datathon is open to all individuals or teams of up to 4 participants; At least half of each team must be individuals who identify as women.
- **Submission Limits (Click here for more details)**
  - You may submit a maximum of 15 entries per day.
  - You may select up to 2 final submissions for judging.
- **Competition Timeline (Click here for more details)**
  - Entry and Team Merger Deadline: February 26, 2023 11:59 PM UTC
  - End Date: March 1, 2023, 2022 11:59 PM UTC
  - Winner will be announced on March 8, 2023.



Group Assignment & Hands-on exercise

- We will use breakout rooms to divide you into groups
- A dedicated mentor will come to your rooms to help you with team formation.
- This is the end of the workshop, you do not need to come unless you have any questions or needs our assistance for group exercise. :) We will stay here for a bit longer to answer any questions about the event.



Group Assignment & Hands-on exercise

Hands-on exercises resources:

- **Colab notebook for code demo on github**
- **Instruction document on github** covers how to use Google colab notebook, how to access all the slides and notebooks and how to attend Kaggle competition.

Remember to register for the competition here!

Top tips from 2 Kaggle Wizards in our team

Dirk Nachbar
Head of Applied Data Science
gTech

Alessandro Mariani
Applied Data Scientist
gTech

How to do well in competitive ML?

1. Invest time.
2. Learn from others (improve on other people's solutions).
3. Be explorative, spend good time on engineering features. Do some reading on the domain.
4. Try many diverse approaches, later you can average/ensemble them.
5. Start simple (linear) and then add complexity incrementally.
6. Partition your data to mirror the competition setup. Use the loss/objective function to select models.
7. If the data is big and slow, play with samples.

WOMEN IN DATA SCIENCE
@ GOOGLE

Tips to succeed on Kaggle (advice from Ale & Dirk*)

How to do well in competitive ML?

1. Invest time.
2. Learn from others (improve on other people's solutions).
3. Be explorative, spend good time on engineering features. Do some reading on the domain.
4. Try many diverse approaches, later you can average/ensemble them.
5. Start simple (linear) and then add complexity incrementally.
6. Partition your data to mirror the competition setup. Use the loss/objective function to select models.
7. If the data is big and slow, play with samples.

Others

1. Perseverance (invest time) - I used to dedicated ~3 hours a day when competing
2. Understand what you're doing - forget the competition and focus on learning how trees, neural networks and linear models works and how you need to prepare data differently. You can come back to competition later.
3. You need to have a solid cross-validation setup to understand if your experiments works (don't rely on public leaderboard feedback, this is how I won my first competition!)
4. Team up - but don't team up before you ran out of ideas! Teaming up is great way to share what each other learnt
5. Read the forum - especially past competitions! Threads are full of knowledge

*Both top 15 before Kaggle got really big :)